# Chapter 6

# Community Enhanced Link Prediction in Dynamic Networks

This Chapter deals with a novel approach community enhanced framework to predict missing links on dynamic social networks[1]. First, a link prediction framework is presented to predict missing links using parameterized influence regions of nodes and their contribution in community partitions. Then, a unique feature set is generated using local, global, and quasi-local similarity-based as well as community information-based features. This feature set is further optimized using scoring-based feature selection methods to select only the most relevant features.

## 6.1 Introduction

The simplest link prediction indices are similarity-based indices, which produce a similarity score S(x,y) for each pair of x and y. The score S(x,y) is determined by the structural or node characteristics of the pair under consideration. Scores based on these

[1]Published in ACM, Transactions on the Web, Community Enhanced Link Prediction in Dynamic Networks

properties can be grouped into local, global, and quasi-local similarity measures. These methods include common neighbors (CN), Adamic/Adar index (AA), Jaccard Coefficient (JC), Preferential Attachment (PA), and so on. Global similarity-based methods consider the whole network topological structure rather than local information, so computational complexity is very high as compared to local similarity indices. Some examples of global similarity measures are Shortest Path (SP), Cos+ (COSP), Matrix Forest Index (MFI), and Average Commute Time (ACT). Quasi-local similarity-based methods achieve a trade-off between both local and global information to improve quality (compared to local measures) and decrease complexity (compared to global measures) of prediction. Some examples are Local Path Index (LP), Path of Length-3 (L3), Clustering Coefficient based Link Prediction index (CCLP) and Node and Link Clustering Coefficient (NLC).

Feature-based link prediction is performed on dynamic networks by dividing the dynamic network into separate snapshots and then using similarity-based link prediction algorithms to create features for current, and possible edges [147]. But this approach has a fundamental flaw in that relations between nodes are not examined in terms of communities to which the nodes belong. It has been suggested in the literature that nodes belonging to the same community have a higher chance of forming a link between themselves [40]. The method proposed by Singh et al. [40] creates communities using information diffusion-based label propagation and then performs link prediction using this community information. The method is shown to be significantly different from both local and quasi-local similarity-based link prediction using statistical tests. One possible reason is that community detection in itself becomes a social network mining method in which both local and global information is referenced to find communities in a graph. Using this reasoning, in this paper, we propose a link prediction feature set that uses features having characteristics of both traditional similarity-based methods and ones having community information-based quantification.

The primary motivation behind the framework of link prediction proposed in this work is

two fold: 1) proposing a similarity calculation method that uses community information to create a feature that has references both local and global information, 2) studying different community detection methods and comparing their effect to the overall link prediction problem on dynamic networks. In this paper, we utilize two different classes of features, i.e., traditional similarity-based (twelve features) and community information-based (eight features). We combine these features to create a feature set that gives enhanced results compared to conventional link prediction algorithms. These feature sets are used to train four different machine learning models for classification - Neural Network (NN), XGBoost (XGB), Linear Discriminant Analysis (LDA), and Random Forest-based Classifier (RFC). This enhanced feature set (also called $COMMLP - FULL$ in this work) is then optimized using feature selection such that features with low significance are dropped. We used feature scoring based on three distinct approaches for this feature selection - tree classifier-based scoring, mutual information-based regression and F-regression estimation. The result is an optimized feature set (also called $COMMLP - DYN$ and $COMMLP$ in this paper) whose performance is compared to three state-of-the-art algorithms [131, 132, 147]. We performed experiments using six different datasets with three evaluation matrices and these experiments indicate that our proposed technique considerably enhances performance.

The main contributions of this paper are as follows:

- We propose a community information-based feature estimation and link prediction method applied to dynamic graphs in a per snapshot feature estimation-based setting.
- We compare and contrast the relevance of eight different community detection methods for community information-based feature estimation. This highlights how some classes of community detection algorithms are more suitable for the link prediction task and others that should be avoided for such frameworks.

- These community information-based are used in combination with other local, global, and quasi-local similarity-based features for enhancing link prediction in dynamic networks.

- The enhanced feature set ($COMMLP-FULL$) is compared individually with twelve classical link prediction algorithms (four local, four global, and four quasi-local similarity-based), which shows the improvement in performance in the per snapshot feature estimation and machine learning classification based setting.

- We perform feature selection on the $COMMLP-FULL$ feature set to create $COMMLP-DYN$, which only contains the most relevant features for link prediction. Finally, $COMMLP-DYN$ is compared with three state-of-the-art algorithms [131, 132, 147], which shows its improved performance.

- All experiments are performed using six datasets and three separate $Ratio$ values (representing the training edges to total edges percentage for training testing set creation). Four different machine learning algorithms are used to estimate better the effect of feature changes on the overall link prediction process.

### 6.1.1 Community Detection

A community can be defined as a deeply linked group of entities. Communities are often used as a form of abstraction where an extremely large sparse graph is converted into strongly connected smaller sub-graphs to make graph processing more manageable. These communities are used in different domains for different purposes [157]. Several approaches have been proposed for community detection in the past few decades, which can be broadly classified into four categories [13] -

- *Modularity optimization-based approaches.* These approaches mainly focus on identifying communities based on density whose quality can be estimated using modularity. Modularity is defined as the difference of connections between nodes

within a community to the total number of connections of the same nodes. Newman [158] introduced the concept of using modularity for community detection, which was later extended by Yu and Ding [159] using spectral clustering and by Agarwal and Kempe [160] using heuristic search.

- *Information theory-based approaches.* The dense communities in this approach are found by identifying sets of nodes sharing some standard features. These approaches can be seen as being similar to clustering methods of the data mining domain. This concept has been combined with random walks [161] as well as node vectors [162] and semantic information [163].

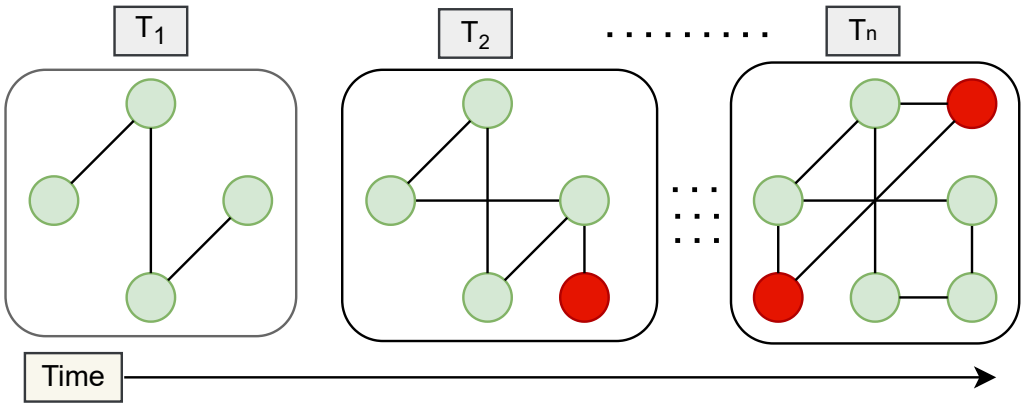
- *Network topology-based approaches.* These approaches are based on overall node arrangement in a network, such as isolated network components are created by removing sparse bridges that connect these components. Centralities and vertex similarities can also be used. Newman [164] proposed an edge betweenness and split betweenness-based algorithm to find communities. Xie and Szymanski [165] and Zhang et al. [166] proposed another such algorithm based on vertex similarity.

- *Hierarchical structure-based approaches.* These approaches are based on similarity measures such that nodes are grouped based on the presence or absence of such measures. Raghavan et al. [167] proposed one such algorithm based on label propagation, while Zhang et al. [168] proposed another such method based on graph diffusion. Xun et al. [169] proposal was based on latent community discovery, while Wei et al.'s [170] proposal was based on spectral clustering and random walks.

### 6.1.2 Community-based Feature Generation Methods

Community detection methods are used to divide a graph into partitions such that intra-community similarity is minimized and inter-community similarity is maximized.

FIGURE 6.1: Structure of snapshots of a dynamic network at different time intervals (green nodes are nodes common with previous snapshot and red nodes are the nodes being added in current snapshot)




Different measures are usually used to calculate the fitness of a community such as modularity, surprise, significance, etc. The following are examples of community detection methods used in this paper.

- **Diffusion Entropy Reducer (DER).** Kozdoba and Mannor [171] proposed a node embedding based community detection algorithm. The algorithm uses random walks to embed the graph in a space of measures. This step produces node embeddings, on which a modification of k-means is used to predict communities. The algorithm is highly distrbutable and the node embeddings, which preserve the network structure in a low dimensional space, can be further used for other purposes such as influence maximization.

- **Surprise communities (SURP).** This community detection method was proposed by Traag et al. [172], where the authors have proposed a new method called surprise to estimate the quality of network partitions. They propose an approximation method to optimize surprise. This surprise is free of the limit of identifying small communities which plagues modularity based algorithms, and hence is more discriminate in identifying communities for sparse networks.

- **Stochastic Block Model (SBM).** This method was proposed by Peixoto et al. [173] in which communities of a graph are identified using a stochastic block model. This

method is especially useful in identifying smaller communities which are removed from consideration in greedy hierarchical structure based methods. This method is based on the principle of parsimony, which filters noise from the overall model, thus preventing the identification of noise based communities.

- **LEIDEN.** This community detection method was proposed by Traag et al. [174] which gives communities with guarantee of connections between them. Also, it provides convergence guarantee for partitions such that all subsets of all communities are locally optimized. The authors have modelled this method to address anamolies in the Louvain algorithm and this algorithm works better than and also in a more time efficient manner than Louvain.

- **Significant communities (SIGNI).** This community detection method was proposed by Traag et al. [175] in which the underlying modular structure of a graph is exploited for community detection. This method also incorporates the concept of significance for detecting thresholds of community detection. As a result, the final communities can be said to give the greatest gain in overall encoding of the graph. Significance can viewed as an alternate to modularity for estimation of community fitness.

- **Constant Potts Model (CPM).** Traag et al. [176] proposed a "resolution-limit-free" method (unlike standard modularity based methods) where instead of being compared to a random null model, the whole graph is compared to a constant factor. The main task of community detection is divided into optimizing internal and external densities with a boundary variable. The communities itself can be considered to be independent from the actual graph.

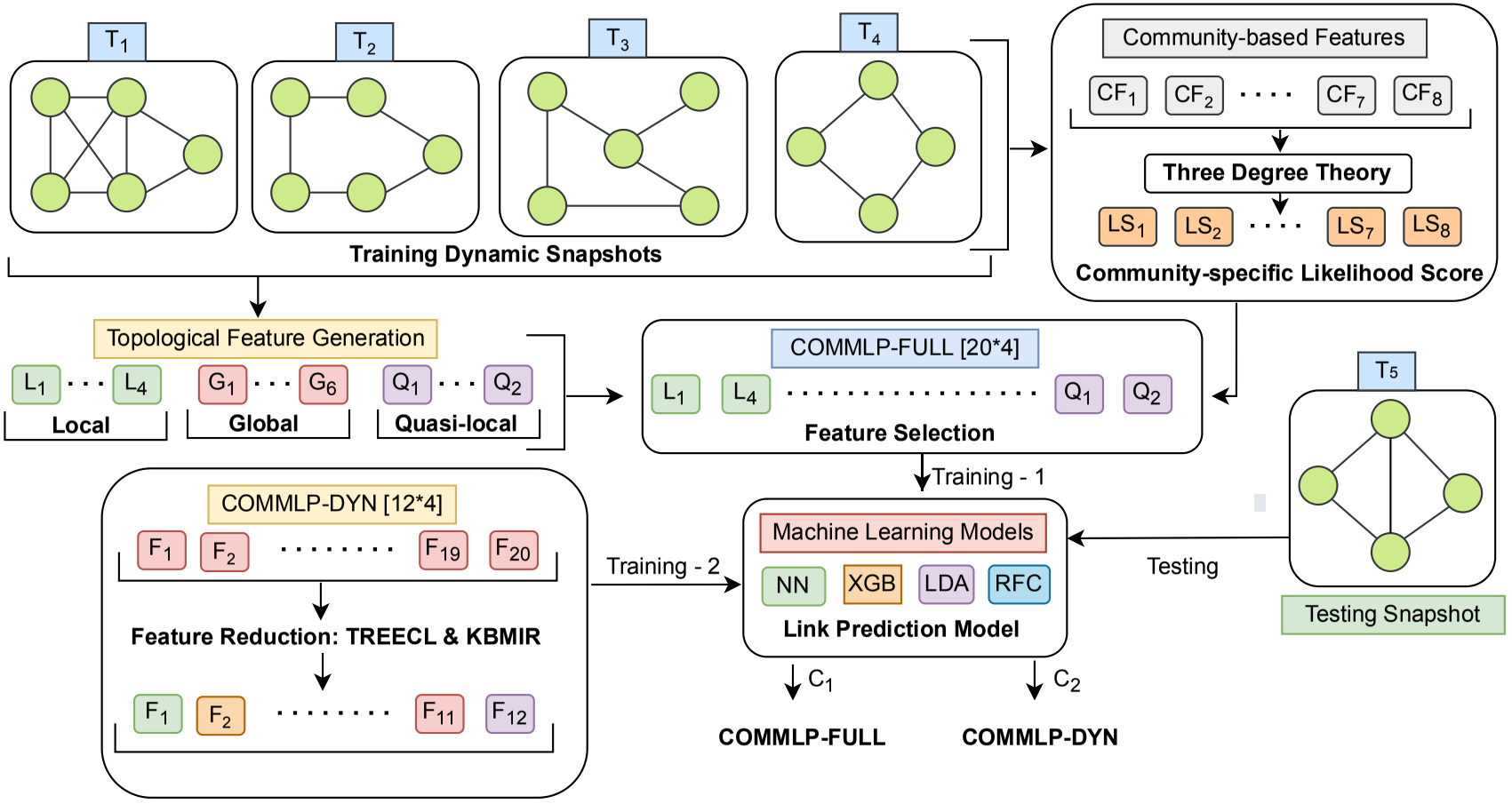
- **Eigenvector (EIGEN).** Newman [177] proposed a recursive algorithm which works on the principle of modularity maximization. This maximization is carried out on eigenvectors of the modularity matrix, almost like the process of calculating Laplacian matrix for graph partitioning. This process can also be viewed as greedy modularity optimization of eigenspectrum.

- **Greedy Modularity (GREED).** Clauset et al. [178] proposed this method which is a hierarchical community detection algorithm, i.e., at each step of iteration, two communities with highest contribution to the global modularity are merged to create a bigger community till a certain threshold is achieved across all communities.

### 6.1.3 Classification Models

To make use of network topological properties and attribute information, the link prediction problem is considered as a learning-based model. A vertex-pair in the network corresponds to a point (training data), and the label of the point indicates whether or not there is an edge (connection) between the vertex-pairs. The problem is treated as a supervised classification model, in which a point (training data) corresponds to a network vertex-pair and the label of the point denotes the existence or absence of an edge (connection) between the pair. This is usually a binary classification task in which many classifiers (e.g., Neural Network (NN), XGBoost (XGB), Linear Discriminant Analysis (LDA), and Random Forest Classifier (RFC)) are used to predict the label of unknown data points (corresponding to missing links in the network) [147, 179, 180]. The selection of relevant feature sets is one of the primary concerns of the machine learning approach [181]. The vast majority of existing research extracts feature sets from network topology (i.e., topological information of the network) [77, 182]. These features are general and cross-domain, and they can be used in any network. Common neighborhood and path-based features are examples of such features. Other research focuses on extracting node and edge information, which is important for improving link prediction performance [77, 183].

FIGURE 6.2: The Proposed Model Framework.The Proposed Model Framework. First, 4 dynamic network snapshots $T_1, T_2, T_3, T_4$ are used to generate 12 topological features based on local, global, and quasi-local information. Also, 8 community-based features are generated based on the three-degree theory. Then we have utilized these 20 features (12 topological and 8 community-specific) for training machine learning models NN, XGB, LDA, and RFC for the COMMLP-FULL prediction model. We have also presented another version of the proposed solution *COMMLP – DYN*, by applying feature reduction methods TREECL, KBREG & KBMIR.



## 6.2 Proposed work

This section discusses the proposed framework $COMMLP-DYN$, which proposes a community-enhanced feature set for link prediction in dynamic networks. The outline of our proposed feature set generation is shown in Fig. 6.2. The figure outlines the feature generation process from snapshots of the dynamic network. We propose two broad classes of features - topological similarity-based (local $L_1,...,L_4$, global $G_1,...,G_4$ and quasi-local $Q_1,...,Q_4$) as well as community information-based ($LS_1,...,LS_8$). First, four dynamic network snapshots $T_1,T_2,T_3,T_4$ are used to generate twelve topological features based on local, global, and quasi-local similarity indices. Also, eight community-based features are generated based on the three-degree theory. Then we have utilized these twenty features (twelve topological features and eight community-specific features) for training machine learning models NN, XGB, LDA, and RFC for the $COMMLP-FULL$ prediction model. We have also presented another version of the proposed solution $COMMLP-DYN$ by applying feature selection methods TREECL, KBMIR & KBREG to optimize the feature to enhance the link prediction accuracy. The community information-based features are derived using 3-degrees-of-influence theory [184, 185] on the community partitions of the graph. The 3-degrees-of-influence theory states that the influence of a central node can be perceived at a maximum distance of three hops from it. Using this theory, we find two nodes' common regions of influence. This common region of influence is then evaluated using community partitions for a quantitative measure of rigidity. This is the community information-based link prediction (likelihood) score. The feature set which contains all features, i.e., four local ($L_1,...,L_4$), four global ($G_1,...,G_4$), four quasi-local ($Q_1,...,Q_4$) similarity-based and eight community information-based ($LS_1,...,LS_8$), is termed as $COMMLP-FULL$. $COMMLP-DYN$ is the optimized form of the $COMMLP-FULL$ feature set that is created after performing feature scoring-based elimination.

### 6.2.1 Topological Feature Generation

Topological similarity-based features are features which are generated directly from snapshots of the dynamic graph using traditional link prediction algorithms. Most of the features we have selected have been used in some combination in the literature [131, 147, 179, 180] and have proved to be highly successful for link prediction in dynamic networks. The local similarity-based methods are Common Neighbors (CN), Adamic/Adar Index (AA), Jaccard Coefficient (JC), and Preferential Attachment (PA), which are represented as $L_1, L_2, L_3, L_4$ respectively. Global similarity-based methods are Shortest Path (SP), Cos+ (COSP), Matrix Forest Index (MFI), and Average Commute Time (ACT), which are represented as $G_1, G_2, G_3, G_4$ respectively. Quasi-local similarity-based methods are Local Path Index (LP), Path of Length 3 (L3), Clustering Coefficient based Link Prediction (CCLP) and Node and Link clustering coefficient (NLC), which are represented as $Q_1, Q_2, Q_3, Q_4$ respectively.

### 6.2.2 Community-based Feature Generation

In this section, we provide the methodology of generating community information-based link prediction features. For this we take the community detection algorithm as input parameter and generate different sores for different algorithms accordingly. The calculation of community information-based link prediction features can be broadly divided into three parts.

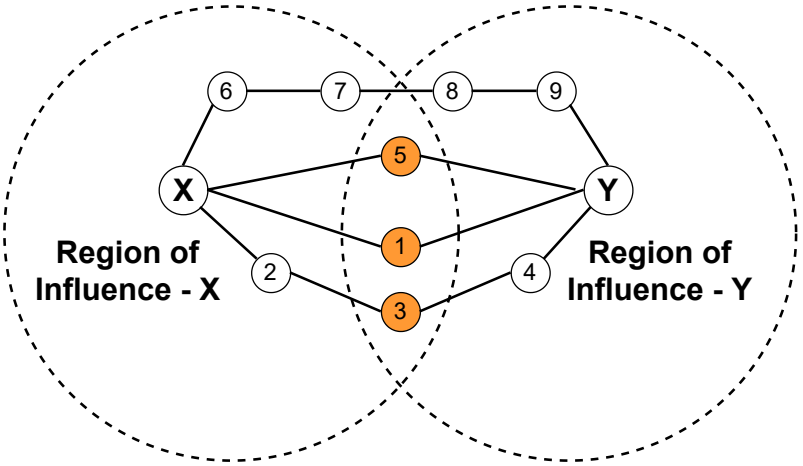
- **Community Detection.** Using some common community detection algorithms described in Section 6.1.2, we divide the graph into separate non-overlapping communities. A point to be noted here is that all nodes are considered to be part of some community, even if the community has a single node as its membership. The partitions are represented as $NCL_1, ...NCL_m$ (node community labels) where $m$ is the total number of partitions. The nodes are related to the community labels using

the notation $CL(x) = NCL_m$, where the community label of node $x$ is $NCL_m$. A further extension to this notation is the type of community detection algorithm used to generate communities on graph, i.e., $CL_{COM}(x) = NCL_m^{COM}$, where $COM$ is the community detection method from DER, SURP, SBM, LEIDEN, SIGNI, CPM, EIGEN, & EIGEN (explained in Section 6.1.2). For example, $CL_{SBM}(x) = NCL_4^{SBM}$, represents that when stochastic block model-based community detection was used on graph, node $x$ was assigned node label $NCL_4$. The total nodes in each community are also stored in dictionary with notation $NodeCount(NCL_1)$ which represents total number of nodes with community label $NCL_1$ and $NodeCount(CL_{COM}(x))$ is number of nodes with same label (same community) as node $x$ when using $COM$ community detection algorithm.

- **Common Influence Region Estimation.** The 3-degrees-of-influence theory [184, 185] states that the influence of a central node can be perceived at a maximum distance of three hops from it. Using this theory, we find two nodes' common region of influence. The node neighborhoods are represented by notation $HOP-n(X)$ where $n$ is the number of hops the other node is maximum distant from node $X$. We can take an example of nodes $X\&Y$ in Fig. 6.3. Here $HOP-1(X)$=$\{1,2,5,6\}$ and $HOP-2(X)$=$HOP-1(X) \cup \{3,7\}$. Similarly, $HOP-1(Y)\&HOP-2(Y)$ can be defined. In this figure based example we assume that influence of a central node extends to 2 hops from it, then common influence region of nodes $X\&Y$, $CIR(X,Y)$=$HOP-2(X) \cap HOP-2(Y)$=$\{1,3,5\}$. In actual formulation and experiments, by 3-degrees-of-influence theory, $CIR(X,Y)$=$HOP-3(X) \cap HOP-3(Y)$.

- **Link Likelihood Score Estimation.** The link likelihood estimation part of our proposal is based on the rigidity of the common influence region between two nodes. We can infer how much effect the nodes of the same community have on the common influence region through the rigidity concept. Hence we estimate the normalized effect of each community on the $CIR$, using the total number of nodes in each community as a normalization factor. One special case is introduced,

FIGURE 6.3: Example of Common Influence Region of nodes $X\&Y$ (represented as orange color nodes 1,3,5) when influence is assumed to be spread to 2-hop region from the central node - $CIR(X,Y)$.




which skips the normalization step. This is the case when community node labels of the two nodes between which likelihood is to be estimated match with the community node label of the node within the common influence region. Equation 6.1 gives the exact calculation procedure for community information-based link prediction measure (for community detection method *COM*).

$$LS(x,y)^{COM} = \sum_{n \in CIR(x,y)} \begin{cases} 1, & iff\ CL_{COM}(x) == CL_{COM}(y) \\ & == CL_{COM}(n) \\ \frac{1}{NodeCount(CL_{COM}(n))}, & otherwise \end{cases} \tag{6.1}$$

This equation represents the main additional complexity of our approach. After community detection of the entire graph, the determination of common influence region between two nodes ($CIR(x,y)$) determines the overall link prediction score. The common influence region calculation involves matching 3 Degree of Influence regions of of nodes $x\&y$. Assuming that determination of common neighbors between two nodes depends on the average degree of graph ($D_{avg}$) such that its

complexity is $O(D_{avg})$ (matching two neighbor lists of $D_{avg}$ length), the overall complexity of calculating common influence region would be $O(D_{avg}^3)$.

### 6.2.3 Feature Set Engineering and Link Prediction

The combination of local, global, quasi-local similarity-based and community information based features ($L_1, L_2, L_3, L_4$, $G_1, G_2, G_3, G_4$, $Q_1, Q_2, Q_3, Q_4$, $LS_1, LS_2, LS_3, LS_4, LS_5, LS_6, LS_7, LS_8$) is the enhanced feature set $COMMLP - FULL$. The steps for using this enhanced feature set for link prediction are as follows:

- In the first step, the overall graph's edge list is utilized as input. It's made up of the network's whole edge list, which includes source and target nodes as well as a timestamp for when the edge happened.
- We divide the entire edge list into equal-time interval snapshots ($G_0, G_1, ..., G_n$). In order to make the time difference between each snapshot nearly equal, we divide the dataset's whole time range (the moment the first and last edges appear) into equal-sized pieces. Each snapshot features edges that are exclusive to this time period.
- For our analysis, we used five snapshots. We combine the most recent snapshot with the randomized non-existing edges to create our training and testing edge lists. To better measure performance change of algorithm with associated information, the combined set of true and non-existing edges is randomly separated into training and testing edge sets, with the ratio of a number of training edges to all edges ranging between 0.7&0.9.
- The following phase entails developing edge features based on snapshots for each edge of the training and testing edge sets. The feature set is based on local similarity-based features, global similarity-based features explained in Section, quasi-local similarity-based features explained in Section 6.1, and our newly

proposed community information-based features presented in Section 6.2.2 which makes a total of twenty features.

- Depending on whether an edge exists in the training or testing subset, we divide the data into training and testing sets. Five snapshots were taken in total, with the class label being determined from the fifth snapshot. Four of the snapshots were used to create features.

- The training data is then fed into machine learning models, which consist of Neural Network (NN), XGBoost (XGB), Linear Discriminant Analysis (LDA), Random Forest Classifier (RFC). These models are then used to predict the probabilities of existing edges on the test subset.

- Finally, we use three performance matrices to contrast our proposed *COMMLP* approach against state-of-the-art algorithms.

### 6.2.4 Feature Reduction

Feature reduction is used to enhance the performance of $COMMLP-FULL$ feature set by eliminating less relevant features. The final feature set created after scoring-based feature elimination from $COMMLP-FULL$ is called $COMMLP-DYN$. The feature relevance scores have been calculated using two separate techniques.

- **Extra Trees Classifier (TREECL)** This method is a meta estimator that employs averaging to increase prediction accuracy and control over-fitting by fitting a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset [156]. The total decrease (normalized) of the criteria brought by a characteristic is used to calculate its significance. It's also known as Gini's significance.

- **Mutual information (KBMIR).** A non-negative quantity that reflects how reliant two random variables are on each other is called mutual information [186, 187]. It

calculates how much information can be gleaned from one random variable given another. It is 0 if and only if two random variables are independent, whereas higher values suggest greater dependency. Non-parametric approaches based on entropy estimates from k-nearest neighbors distances are used for scoring. Under the condition of target estimation, it may be stated to quantify entropy reductions.

- **F-regression (KBREG)**. If every feature is positively correlated with the target, $F-regression$, which is derived from $R-regression$, will rank the features in the same order [156]. However, please note that $r\_regression$ values, in contrast to $F-regression$, are in the [-1, 1] range and might thus be negative. Therefore, regardless of the direction of the connection with the target variable, $F-regression$ is advised as a feature selection criterion to uncover potentially predictive features for a downstream classifier.

### 6.2.5 Algorithm Description

The **Algorithm 5** demonstrates how the likelihood of possible edges is calculated using our proposed $COMMLP-DYN$ approach. The input to the algorithm is the dynamic graph in snapshot form $G_i \mid i \in (1,n)$. The output is the likelihood probability of possible edges using both $COMMLP-FULL$ and $COMMLP-DYN$ feature sets. The algorithm can be divided into three major modules - initialization (lines 1-3), feature generation and reduction (4-14), and training and testing machine learning model phase (lines 15-22). We create graphs from snapshots (line 1) in the initialization phase and generate training and testing edge sets (lines 2,3). In lines 4-8, we calculate the similarity-based link prediction scores for training edges. In lines 9-12, we calculate community information-based link prediction scores for training edges. Line 12 is the concatenation of all features to create $COMMLP-FULL$ feature set, and line 14 is feature scoring based reduction of $COMMLP-FULL$ to create $COMMLP-DYN$. Lines 15-17 train machine learning-based classification models using these feature sets created on the

**Algorithm 5:** COMMLP-DYN: Community Enhanced Link Prediction Algorithm for Dynamic Networks

**Input:** Social Networks: $G_i(V_i, E_i)$, $0 \leq i \leq 4$, Ratio: Training edges to total edges percentage

**Output:** Likelihood Score of Possible Links

1 Generate graphs $[T_1, \ldots, T_5]$ for different snapshots $[S_1, \ldots, S_4]$ for training and testing.

2 $E_{TOTAL} \leftarrow E_4 \cup$ *non edges of 4-th snapshot* ▷ Possible true and false edges

3 $E_{TRAIN}, E_{TEST} \leftarrow E_{TOTAL}$ ▷ Possible edges randomly divided into training and testing sets on basis of Ratio variable

4 ▷ Calculating topological features

5 **for** *each edge* $(x, y) \in E_{TRAIN}$ **do**

6 $\quad L_{i=1..4}(x, y) \leftarrow \{CN, JC, AA, PA\}$ ▷ Local Features

7 $\quad G_{i=1..4}(x, y) \leftarrow \{SP, COSP, MFI, ACT\}$ ▷ Global Features

8 $\quad Q_{i=1..4}(x, y) \leftarrow \{LP, L3, CCLP, NLC\}$ ▷ Quasi-Local Features

9 $[CF_1, CF_2, \ldots, CF_8] \leftarrow$ Identify node community labels using eight community detection methods DER, SURP, SBM, LEIDEN, SIGNI, CPM, EIGEN, and EIGEN. ▷ Community Label

10 **for** *each edge* $(x, y) \in E_{TRAIN}$ **do**

11 $\quad$ **for** *i= 1 to 8* **do**

12 $\quad\quad LS_i(x, y) \leftarrow$ *Community Information-based Features* ▷ Using Eq. 6.1

13 $COMMLP - FULL \leftarrow [L_1, \ldots, L_4, G_1, \ldots, G_4, Q_1, \ldots, Q_4, LS_1, \ldots, LS_8]$ ▷ Full feature-set

14 $COMMLP - DYN \leftarrow$ Generate a reduced feature-set $[F_1, F_2, \ldots, F_{12}]$ using feature scoring based on TREECL and KBMIR methods

15 **for** $i \in \{NN, XGB, LDA, RFC\}$ **do**

16 $\quad Model - Full^i \leftarrow Fit(COMMLP - FULL)$ ▷ Training model on full feature-set

17 $\quad Model - Opti^i \leftarrow Fit(COMMLP - DYN)$ ▷ Training model on reduced feature-set

18 ▷ Creation of $COMMLP - FULL\&COMMLP - DYN$ feature sets for testing edge set $E_{TEST}$

19 **for** *each* $(u, v) \in E_{TEST}$ **do**

20 $\quad$ **for** $i \in \{NN, XGB, LDA, RFC\}$ **do**

21 $\quad\quad LS^i_{FULL}(u, v) \leftarrow Model - Full^i(PredictProbab(u, v))$ ▷ Probability prediction based on full feature-set

22 $\quad\quad LS^i_{OPTI}(u, v) \leftarrow Model - Opti^i(PredictProbab(u, v))$ ▷ Probability prediction based on reduced feature-set

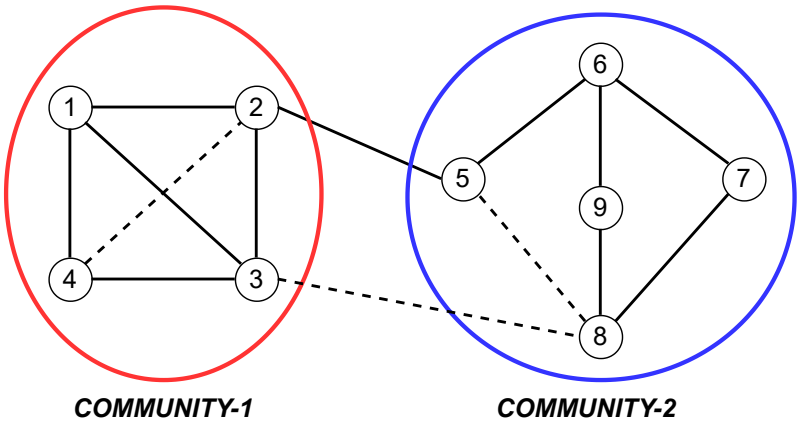
23 **Return** $LS^i_{FULL}, LS^i_{OPTI}$;

training edge set. In line 18, we create $COMMLP - FULL\&COMMLP - DYN$ for the testing edge set using the same procedure for the training edge set (lines 4-13). In lines

19-22, we estimate probabilities of testing edges using both feature sets, $COMMLP-FULL\&COMMLP-DYN$.

### 6.2.6 Demonstration with Example

FIGURE 6.4: Example Graph for Feature Calculation




In this section, we provide an example graph and demonstrate the calculation process for the proposed community-based link prediction feature by using Equation 6.1. Fig. 6.4 provides the demonstrative graph we use for calculations. This graph has nine nodes numbered from $1-9$ divided into two assumed community partition sets, $\{1,2,3,4\}$ and $\{5,6,7,8,9\}$. The true edges of the graph are represented with solid lines, and dashed lines represent the non-edges we will use in our example. Table 6.1 provides this graph's localized edge features of all true and non-edges. Columns labelled *CN*, *JC*, *PA*, *AA*, *SP*, *L3*, *CCLP*, and *NLC* contain the feature values calculated using classical link prediction algorithms. The column labeled $COM-DYN$ gives value for the proposed community-based link prediction feature, and other columns provide the information required for its calculation. First, let us consider intra-community non-edges, i.e., possible edges inside the same community. Let us take the example of a possible edge between nodes 2&4 (shown with a dashed line inside $COMMUNITY-1$). Since this is an intra-community edge, the labels of the endpoints ($CL-COM(2)$ and $CL-COM(4)$) are the same. Also the nodes within 3-hop region of influence of nodes 2&4 are,

$HOP-3(2)$=$\{1,3,4,5,6,7\}$ and $HOP-3(4)$=$\{1,2,3,5\}$. The common influence region of nodes 2&4 ($CIR(2,4)$) is set $\{1,3,5\}$. This $CIR$ has been calculated using methodology explained in Section 6.2.2 and Fig. 6.3. The $CIR$s and community labels for all possible node pairs can be found listed in Table 6.1. In $CIR(2,4)$, nodes 1&3 belong to the same community and hence each of their individual contribution is equal to 1 (using Equation 6.1). For node 5, the second case of Equation 6.1 applied whereby the node contribution becomes $\frac{1}{NodeCount(CL-COM(5))}$ which is equal to 0.2. Hence the $COM-DYN$ for node pair 2&4 becomes 2.2. Let us take another case of node pair 5&8 in $COMMUNITY-2$. The community labels are same and $CIR(5,8)$ is the set $\{6,9,7\}$ (from Table 6.1). Since all nodes of this set belong to $COMMUNITY-2$ itself then $COM-DYN$ for this node pair would be 3. Now let us take example of edge pair 3&8 which is an inter-community edge, i.e., the end points of this edge belong to different communities. The $CIR(3,8)$ set for this pair is $\{5,6\}$ (from Table 6.1). Since both nodes 5&6 belong to $COMMUNITY-2$ and number of nodes in this community is 5 then $\frac{1}{NodeCount(CL-COM(5))} = \frac{1}{NodeCount(CL-COM(6))} = 0.2$. Hence the $COM-DYN$ for inter-community node pair 3&8 would be 0.4. Apart from the calculation of $COM-DYN$ features, the formulae for calculation of other link prediction methods shown in Table 6.1. Taking the case of non-edges discussed above, $CN(2,4)$=$|\Gamma(2) \cap \Gamma(4)|$=$|\{1,3,5\} \cap \{1,3\}|$=$|\{1,3\}|$=2 and $PA(2,4)$=$|\Gamma(2)| * |\Gamma(4)|$=$|\{1,3,5\}| * |\{1,3\}|$=$3*2$=6. Here, $\Gamma(x)$ represents immediate neighbor set of node $x$. We can make a final observation from Table 6.1 by noting that in many cases of non-edges, immediate common neighbor-based similarities, as well as path and clustering-based similarities, do not provide much information for prediction (value 0 in non-edges between nodes 1&6, 1&7, 1&8, etc.). Even in those cases, the $COM-DYN$ feature provides some measure for the possibility of edge existence.

TABLE 6.1: Scoring of different link prediction methods in comparison to our proposed feature in example graph ($CIR(X,Y)$ is Common Influence Region between $X\&Y$, $CL-COM(X)\&CL-COM(Y)$ are community labels of $X\&Y$ and $COMMLP-DYN$ is the feature calculated using this information)

| Category | Edge(X-Y) | CN | JC | PA | AA | SP | L3 | CCLP | NLC | CIR(X,Y) | CL-COM(X) | CL-COM(Y) | COM-DYN |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TRUE EDGES | 1 - 2 | 1.0 | 0.2 | 9.0 | 0.9 | 1.0 | 2.2 | 0.7 | 1.0 | 3, 4, 5, 6 | 1 | 1 | 2.40 |
| | 1 - 3 | 2.0 | 0.5 | 9.0 | 2.4 | 1.0 | 1.8 | 1.3 | 2.3 | 2, 4, 5, 6 | 1 | 1 | 2.40 |
| | 1 - 4 | 1.0 | 0.3 | 6.0 | 0.9 | 1.0 | 1.8 | 0.7 | 1.0 | 2, 3, 5 | 1 | 1 | 2.20 |
| | 2 - 3 | 1.0 | 0.2 | 9.0 | 0.9 | 1.0 | 2.2 | 0.7 | 1.0 | 1, 4, 5, 6 | 1 | 1 | 2.40 |
| | 2 - 5 | 0.0 | 0.0 | 6.0 | 0.0 | 1.0 | 1.5 | 0.0 | 0.0 | 1, 3, 4, 6, 7, 9 | 1 | 2 | 1.35 |
| | 3 - 4 | 1.0 | 0.3 | 6.0 | 0.9 | 1.0 | 1.8 | 0.7 | 1.0 | 1, 2, 5 | 1 | 1 | 2.20 |
| | 5 - 6 | 0.0 | 0.0 | 6.0 | 0.0 | 1.0 | 1.6 | 0.0 | 0.0 | 1, 2, 3, 7, 8, 9 | 2 | 2 | 3.75 |
| | 6 - 7 | 0.0 | 0.0 | 6.0 | 0.0 | 1.0 | 2.2 | 0.0 | 0.0 | 8, 9, 2, 5 | 2 | 2 | 3.25 |
| | 6 - 9 | 0.0 | 0.0 | 6.0 | 0.0 | 1.0 | 2.2 | 0.0 | 0.0 | 8, 2, 5, 7 | 2 | 2 | 3.25 |
| | 7 - 8 | 0.0 | 0.0 | 4.0 | 0.0 | 1.0 | 1.8 | 0.0 | 0.0 | 9, 5, 6 | 2 | 2 | 3.00 |
| | 8 - 9 | 0.0 | 0.0 | 4.0 | 0.0 | 1.0 | 1.8 | 0.0 | 0.0 | 5, 6, 7 | 2 | 2 | 3.00 |
| NON EDGES | 1 - 5 | 1.0 | 0.3 | 6.0 | 0.9 | 2.0 | 0.3 | 0.3 | 0.2 | 2, 3, 4, 6 | 1 | 2 | 0.95 |
| | 1 - 6 | 0.0 | 0.0 | 9.0 | 0.0 | 3.0 | 0.4 | 0.0 | 0.0 | 2, 3, 5 | 1 | 2 | 0.70 |
| | 1 - 7 | 0.0 | 0.0 | 6.0 | 0.0 | 4.0 | 0.0 | 0.0 | 0.0 | 2, 5, 6 | 1 | 2 | 0.65 |
| | 1 - 8 | 0.0 | 0.0 | 6.0 | 0.0 | 5.0 | 0.0 | 0.0 | 0.0 | 5, 6 | 1 | 2 | 0.40 |
| | 1 - 9 | 0.0 | 0.0 | 6.0 | 0.0 | 4.0 | 0.0 | 0.0 | 0.0 | 2, 5, 6 | 1 | 2 | 0.65 |
| | 2 - 4 | 2.0 | 0.7 | 6.0 | 1.8 | 2.0 | 0.7 | 1.3 | 1.3 | 1, 3, 5 | 1 | 1 | 2.20 |
| | 2 - 6 | 1.0 | 0.2 | 9.0 | 1.4 | 2.0 | 0.0 | 0.0 | 0.0 | 1, 3, 5, 7, 9 | 1 | 2 | 1.10 |
| | 2 - 7 | 0.0 | 0.0 | 6.0 | 0.0 | 3.0 | 0.4 | 0.0 | 0.0 | 9, 5, 6 | 1 | 2 | 0.60 |
| | 2 - 8 | 0.0 | 0.0 | 6.0 | 0.0 | 4.0 | 0.0 | 0.0 | 0.0 | 9, 5, 6, 7 | 1 | 2 | 0.80 |
| | 2 - 9 | 0.0 | 0.0 | 6.0 | 0.0 | 3.0 | 0.4 | 0.0 | 0.0 | 5, 6, 7 | 1 | 2 | 0.60 |
| | 3 - 5 | 1.0 | 0.3 | 6.0 | 0.9 | 2.0 | 0.3 | 0.3 | 0.2 | 1, 2, 4, 6 | 1 | 2 | 0.95 |
| | 3 - 6 | 0.0 | 0.0 | 9.0 | 0.0 | 3.0 | 0.4 | 0.0 | 0.0 | 1, 2, 5 | 1 | 2 | 0.70 |
| | 3 - 7 | 0.0 | 0.0 | 6.0 | 0.0 | 4.0 | 0.0 | 0.0 | 0.0 | 2, 5, 6 | 1 | 2 | 0.65 |
| | 3 - 8 | 0.0 | 0.0 | 6.0 | 0.0 | 5.0 | 0.0 | 0.0 | 0.0 | 5, 6 | 1 | 2 | 0.40 |
| | 3 - 9 | 0.0 | 0.0 | 6.0 | 0.0 | 4.0 | 0.0 | 0.0 | 0.0 | 2, 5, 6 | 1 | 2 | 0.65 |
| | 4 - 5 | 0.0 | 0.0 | 4.0 | 0.0 | 3.0 | 0.7 | 0.0 | 0.0 | 1, 2, 3 | 1 | 2 | 0.75 |
| | 4 - 6 | 0.0 | 0.0 | 6.0 | 0.0 | 4.0 | 0.0 | 0.0 | 0.0 | 1, 2, 3, 5 | 1 | 2 | 0.95 |
| | 4 - 7 | 0.0 | 0.0 | 4.0 | 0.0 | 5.0 | 0.0 | 0.0 | 0.0 | 2, 5 | 1 | 2 | 0.45 |
| | 4 - 8 | 0.0 | 0.0 | 4.0 | 0.0 | 6.0 | 0.0 | 0.0 | 0.0 | 5 | 1 | 2 | 0.20 |
| | 4 - 9 | 0.0 | 0.0 | 4.0 | 0.0 | 5.0 | 0.0 | 0.0 | 0.0 | 2, 5 | 1 | 2 | 0.45 |
| | 5 - 7 | 1.0 | 0.3 | 4.0 | 0.9 | 2.0 | 0.0 | 0.0 | 0.0 | 8, 9, 2, 6 | 2 | 2 | 3.25 |
| | 5 - 8 | 0.0 | 0.0 | 4.0 | 0.0 | 3.0 | 0.8 | 0.0 | 0.0 | 9, 6, 7 | 2 | 2 | 3.00 |
| | 5 - 9 | 1.0 | 0.3 | 4.0 | 0.9 | 2.0 | 0.0 | 0.0 | 0.0 | 8, 2, 6, 7 | 2 | 2 | 3.25 |
| | 6 - 8 | 2.0 | 0.7 | 6.0 | 2.9 | 2.0 | 0.0 | 0.0 | 0.0 | 9, 5, 7 | 2 | 2 | 3.00 |
| | 7 - 9 | 2.0 | 1.0 | 4.0 | 2.4 | 2.0 | 0.0 | 0.0 | 0.0 | 8, 2, 5, 6 | 2 | 2 | 3.25 |

## 6.3 Result Analysis

In this section, we'll explore the experimental results from six well-known dynamic datasets using three evaluation metrics, AUC, AUPR and AVG PREC. We evaluate and analyze the performance of the proposed *COMMLP* technique individually as well as three state-of-the-art algorithms with four different machine learning models (NN, XGB, LDA, and RFC). As state-of-the-art algorithms, we have selected three feature-based link prediction algorithms. The first algorithm is the one proposed by Mukesh et al. [147], referred to as *LGQ* in this paper. Chiu et al.'s [131] proposal is the second algorithm that is employed in this framework and is referred to as WEAK in this work. The third

state-of-the-art algorithm in this paper is N2V, which is taken from work by De Winter et al. [132].

Additionally, we have evaluated the feature scores of community-based features and individual link prediction features. We have enhanced the proposed approach based on the feature score. In this section, we'll explore the experimental results from six well-known dynamic datasets using three evaluation metrics. The evaluation of performance is accomplished using three different training and testing ratios, which are $0.7, 0.8, \& 0.9$. We employ Python's Scikit-learn package's default settings for our predictive models [156].

### 6.3.1 Performance comparison of $COMMLP-FULL$ with individual feature based methods using Neural Network model

Table 6.2 compares the performance of the $COMMLP-FULL$ feature set with twelve individual link prediction algorithm-based feature sets on five datasets and three *Ratio* values on an Neural Network-based classifier. The learning rate we used to train the model is 0.001. We trained the model for 5 epochs with a batch size of 32. We employed 2 layers, each with 1024 RELU activators, a learning rate of 0.001, 5 epochs, and a batch size of 32 samples for the hidden layer architecture.

With respect to the AUC performance metric, our proposed $COMMLP-FULL$ feature set is the seventh-best performing algorithm behind SP, COSP, MFI, L3, LOCALP, NLC and CCLP. In the AUPR metric, our algorithm performs worse than global similarity-based algorithms, SP, COSP, and MFI. In EU-Core, FB-Forum, and CollegeMsg datasets, it is outperformed by the L3 algorithm and also by NLC and CCLP algorithms only in EU-Core dataset. In the Average Precision (AVG PREC) metric, the performance of our algorithm follows a similar pattern to the AUPR metric, where it is consistently outperformed by global similarity-based features. In EU-Core, FB-Forum, and CollegeMsg datasets, it is outperformed by the $L3$ algorithm and also by NLC and

TABLE 6.2: Performance comparison of *COMMLP − FULL* with individual feature based link prediction algorithms using Neural Network model on five datasets, three *Ratio* values and three performance metrics

| METRIC | DATASET | RATIO | CN | AA | JC | PA | SP | COSP | ACT | MFI | L3 | LOCALP | CCLP | NLC | COMMLP-FULL |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AUC | MIT | 0.7 | 0.68469 | 0.69899 | 0.77959 | 0.63465 | 0.84692 | **0.88152** | 0.64583 | 0.85917 | 0.72721 | 0.64169 | 0.83644 | 0.83425 | 0.77001 |
| | | 0.8 | 0.69049 | 0.72782 | 0.79215 | 0.63174 | 0.85595 | **0.87995** | 0.58485 | 0.84996 | 0.72747 | 0.66896 | 0.82874 | 0.8457 | 0.77153 |
| | | 0.9 | 0.70721 | 0.74606 | 0.80264 | 0.62897 | 0.85849 | **0.88458** | 0.62802 | 0.87379 | 0.73834 | 0.69035 | 0.82573 | 0.84325 | 0.76182 |
| | Radoslaw-Email | 0.7 | 0.70777 | 0.84499 | 0.86514 | 0.53547 | 0.91261 | 0.92877 | 0.55486 | **0.92102** | 0.77418 | 0.69444 | 0.86235 | 0.8657 | 0.80569 |
| | | 0.8 | 0.74475 | 0.81269 | 0.87116 | 0.56788 | 0.91238 | **0.9289** | 0.55797 | **0.92081** | 0.82081 | 0.68438 | 0.86332 | 0.86474 | 0.81507 |
| | | 0.9 | 0.79731 | 0.84907 | 0.86377 | 0.59627 | 0.91277 | **0.92932** | 0.56915 | **0.92131** | 0.80297 | 0.70648 | 0.86092 | 0.86932 | 0.8565 |
| | EU-Core | 0.7 | 0.86891 | 0.92095 | 0.92636 | 0.51857 | **0.97114** | **0.97574** | 0.50402 | 0.95027 | 0.92796 | 0.77869 | 0.93863 | 0.94021 | 0.76136 |
| | | 0.8 | 0.9009 | 0.92916 | 0.92519 | 0.5127 | 0.96975 | **0.97554** | 0.50513 | 0.95159 | 0.9187 | 0.82826 | 0.93932 | 0.93671 | 0.75736 |
| | | 0.9 | 0.89894 | 0.92623 | 0.92766 | 0.51562 | 0.96878 | **0.97304** | 0.51041 | 0.95138 | 0.9157 | 0.81226 | 0.93924 | 0.94069 | 0.76916 |
| | FB-Forum | 0.7 | 0.55408 | 0.5644 | 0.61656 | 0.55891 | **0.91446** | **0.91005** | 0.5223 | 0.89367 | **0.91141** | 0.73432 | 0.57555 | 0.59487 | 0.73321 |
| | | 0.8 | 0.54287 | 0.56725 | 0.61142 | 0.55302 | **0.91793** | **0.91453** | 0.51466 | 0.89661 | 0.90436 | 0.78829 | 0.57164 | 0.59706 | 0.736 |
| | | 0.9 | 0.56669 | 0.57879 | 0.61707 | 0.53981 | 0.91866 | **0.92174** | 0.51877 | 0.89319 | 0.90957 | 0.8121 | 0.57018 | 0.59207 | 0.72083 |
| | CollegeMsg | 0.7 | 0.51166 | 0.5199 | 0.54028 | 0.52116 | 0.65187 | 0.66875 | 0.50222 | 0.6521 | **0.67609** | 0.59507 | 0.52348 | 0.52916 | 0.58835 |
| | | 0.8 | 0.50905 | 0.50999 | 0.54802 | 0.53851 | 0.66512 | **0.67926** | 0.50351 | 0.64392 | **0.67602** | 0.60392 | 0.51977 | 0.53322 | 0.59522 |
| | | 0.9 | 0.50936 | 0.51757 | 0.54684 | 0.52712 | 0.66506 | **0.68063** | 0.49904 | 0.65841 | **0.68841** | 0.58953 | 0.52327 | 0.53026 | 0.58508 |
| AUPR | MIT | 0.7 | 0.501 | 0.47149 | 0.52624 | 0.54096 | 0.5818 | **0.69048** | 0.54426 | 0.62193 | 0.55111 | 0.48843 | 0.58519 | 0.59215 | 0.65801 |
| | | 0.8 | 0.49141 | 0.51605 | 0.53892 | 0.53577 | 0.58656 | **0.67657** | 0.50524 | 0.59205 | 0.57492 | 0.51085 | 0.56151 | 0.59857 | 0.6177 |
| | | 0.9 | 0.49421 | 0.51114 | 0.58781 | 0.58666 | 0.6227 | **0.705** | 0.51234 | 0.65853 | 0.54992 | 0.5383 | 0.59289 | 0.60082 | 0.64055 |
| | Radoslaw-Email | 0.7 | 0.42018 | 0.54867 | 0.6154 | 0.32359 | 0.76404 | **0.80781** | 0.36045 | 0.7532 | 0.48956 | 0.41923 | 0.58251 | 0.60696 | 0.60669 |
| | | 0.8 | 0.43921 | 0.51937 | 0.62244 | 0.3741 | 0.7772 | **0.80595** | 0.35886 | 0.75254 | 0.55028 | 0.43058 | 0.58712 | 0.60044 | 0.63342 |
| | | 0.9 | 0.49963 | 0.56226 | 0.63231 | 0.44024 | 0.78814 | **0.80969** | 0.37109 | 0.7567 | 0.52512 | 0.43791 | 0.57186 | 0.60571 | 0.6475 |
| | EU-Core | 0.7 | 0.63252 | 0.72339 | 0.70075 | 0.2691 | 0.83494 | **0.86568** | 0.32751 | 0.77532 | 0.73604 | 0.54792 | 0.77307 | 0.7863 | 0.69027 |
| | | 0.8 | 0.68222 | 0.73142 | 0.6999 | 0.277 | 0.8066 | **0.87531** | 0.35305 | 0.77432 | 0.73798 | 0.61761 | 0.76855 | 0.77023 | 0.69078 |
| | | 0.9 | 0.67449 | 0.73334 | 0.70312 | 0.25246 | 0.85691 | **0.86683** | 0.36778 | 0.76312 | 0.71305 | 0.57567 | 0.78011 | 0.78258 | 0.69089 |
| | FB-Forum | 0.7 | 0.23604 | 0.23572 | 0.28897 | 0.34226 | 0.77968 | **0.81282** | 0.27565 | 0.75716 | 0.79883 | 0.44685 | 0.26539 | 0.29675 | 0.56551 |
| | | 0.8 | 0.22893 | 0.235 | 0.29331 | 0.35334 | 0.79496 | **0.82456** | 0.26455 | 0.77019 | 0.79486 | 0.47883 | 0.25075 | 0.31219 | 0.57845 |
| | | 0.9 | 0.24798 | 0.25751 | 0.30129 | 0.30099 | 0.79631 | **0.83454** | 0.26457 | 0.7672 | 0.79985 | 0.51243 | 0.24166 | 0.32116 | 0.57168 |
| | CollegeMsg | 0.7 | 0.19085 | 0.20161 | 0.21733 | 0.24904 | 0.33817 | **0.46666** | 0.25069 | 0.428 | 0.45564 | 0.31287 | 0.22385 | 0.25559 | 0.31004 |
| | | 0.8 | 0.19352 | 0.18981 | 0.2258 | 0.27338 | 0.38936 | **0.47378** | 0.26501 | 0.4111 | 0.46401 | 0.31122 | 0.22848 | 0.25622 | 0.31393 |
| | | 0.9 | 0.21617 | 0.20417 | 0.22816 | 0.25792 | 0.3611 | **0.48472** | 0.26495 | 0.4507 | 0.4763 | 0.30845 | 0.23481 | 0.25494 | 0.31995 |
| AVG PREC | MIT | 0.7 | 0.47895 | 0.475 | 0.5303 | 0.35219 | 0.60562 | 0.69247 | 0.45825 | 0.62558 | 0.51742 | 0.41467 | 0.59035 | 0.59699 | 0.63389 |
| | | 0.8 | 0.47782 | 0.5214 | 0.54454 | 0.34235 | 0.61482 | 0.68069 | 0.43884 | 0.59824 | 0.53132 | 0.44894 | 0.56932 | 0.60627 | 0.58608 |
| | | 0.9 | 0.48697 | 0.52179 | 0.59512 | 0.34896 | 0.65403 | 0.71188 | 0.41242 | 0.66779 | 0.53609 | 0.4815 | 0.60312 | 0.61132 | 0.61978 |
| | Radoslaw-Email | 0.7 | 0.4219 | 0.55039 | 0.61647 | 0.24438 | 0.76474 | **0.80795** | 0.34186 | 0.75402 | 0.49041 | 0.41928 | 0.58421 | 0.60869 | 0.60663 |
| | | 0.8 | 0.44112 | 0.52193 | 0.62394 | 0.27512 | 0.76676 | **0.80662** | 0.33702 | 0.75407 | 0.55198 | 0.43077 | 0.5897 | 0.60292 | 0.63428 |
| | | 0.9 | 0.50415 | 0.56682 | 0.63458 | 0.30983 | 0.77134 | **0.81102** | 0.3643 | 0.75877 | 0.52821 | 0.4399 | 0.57659 | 0.60992 | 0.64902 |
| | EU-Core | 0.7 | 0.63225 | 0.7243 | 0.7023 | 0.17697 | **0.86709** | **0.86638** | 0.26053 | 0.7767 | 0.73607 | 0.54252 | 0.77374 | 0.78675 | 0.62766 |
| | | 0.8 | 0.68269 | 0.73286 | 0.70215 | 0.18431 | 0.85946 | **0.87593** | 0.24882 | 0.7762 | 0.73855 | 0.61578 | 0.76983 | 0.77153 | 0.65026 |
| | | 0.9 | 0.67674 | 0.73621 | 0.70799 | 0.18265 | **0.86403** | **0.86793** | 0.25781 | 0.7661 | 0.71529 | 0.57507 | 0.78176 | 0.78432 | 0.62663 |
| | FB-Forum | 0.7 | 0.2244 | 0.23068 | 0.27944 | 0.24487 | 0.78241 | **0.81037** | 0.20009 | 0.75949 | 0.79826 | 0.44841 | 0.25664 | 0.28582 | 0.50824 |
| | | 0.8 | 0.2183 | 0.23053 | 0.28417 | 0.23984 | 0.79679 | **0.82186** | 0.21704 | 0.77205 | 0.79364 | 0.48086 | 0.24327 | 0.30131 | 0.52248 |
| | | 0.9 | 0.23657 | 0.25522 | 0.29672 | 0.23762 | 0.79827 | **0.83354** | 0.21297 | 0.77036 | 0.80069 | 0.51701 | 0.23852 | 0.31215 | 0.51697 |
| | CollegeMsg | 0.7 | 0.18441 | 0.19603 | 0.20595 | 0.1931 | 0.32358 | **0.44837** | 0.1716 | 0.4289 | **0.44545** | 0.31124 | 0.21233 | 0.23037 | 0.22893 |
| | | 0.8 | 0.19135 | 0.18722 | 0.21018 | 0.22016 | 0.36511 | **0.4559** | 0.17128 | 0.41348 | **0.456** | 0.3112 | 0.2172 | 0.23699 | 0.2313 |
| | | 0.9 | 0.19994 | 0.20192 | 0.21698 | 0.2262 | 0.35256 | **0.46978** | 0.17697 | 0.45369 | **0.46645** | 0.31307 | 0.2218 | 0.23579 | 0.26649 |

CCLP algorithms only in EU-Core dataset. We can observe from the Neural Network model that there is a scope for improvement of our algorithm.

### 6.3.2 Performance comparison of $COMMLP-FULL$ with individual feature based methods using XGBoost model

Table 6.3 compares the performance of the $COMMLP-FULL$ feature set with twelve individual link prediction algorithm-based feature sets on five datasets and three *Ratio* values on an XGBoost-based classifier. For this classification, we employed 50 estimators with a learning rate of 0.01.

With respect to the AUC performance metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets and *Ratio* values except one. That is $Ratio = 0.9$ for the CollegeMsg dataset, where it is slightly outperformed by L3 and SP, but the values are still very close. Both SP and COSP algorithms show a consistently high performance across all datasets, and *Ratio* values are only slightly worse than $COMMLP-FULL$. In the AUPR performance metric, our algorithm $COMMLP-FULL$ is the best performing algorithm in MIT and FB-Forum datasets and second-best in other datasets. In Radoslaw-Email, EU-Core, and CollegeMsg datasets, the *SP* algorithm slightly outperforms our $COMMLP-FULL$ algorithm. In the Average Precision (AVG PREC) metric, the performance of $COMMLP-FULL$ follows a similar pattern as to the AUPR metric, where it is the best performing algorithm in MIT and FB-Forum datasets and the second-best algorithm in the rest of the datasets.

TABLE 6.3: Performance comparison of *COMMLP − FULL* with individual feature based link prediction algorithms using XGBoost model on five datasets, three *Ratio* values and three performance metrics

| METRIC | DATASET | RATIO | CN | AA | JC | PA | SP | COSP | ACT | MFI | L3 | LOCALP | CCLP | NLC | COMMLP-FULL |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AUC | MIT | 0.7 | 0.71262 | 0.72376 | 0.68188 | 0.73481 | 0.8207 | 0.81429 | 0.76789 | 0.80698 | 0.76915 | 0.74086 | 0.71217 | 0.73157 | **0.8318** |
| | | 0.8 | 0.72284 | 0.71983 | 0.68777 | 0.75206 | 0.82514 | 0.8369 | 0.76973 | 0.79886 | 0.77157 | 0.75542 | 0.71652 | 0.72937 | **0.84181** |
| | | 0.9 | 0.74346 | 0.71822 | 0.68667 | 0.76895 | 0.81998 | 0.83023 | 0.77915 | 0.82781 | 0.77647 | 0.75561 | 0.72138 | 0.73865 | **0.85035** |
| | Radoslaw-Email | 0.7 | 0.68301 | 0.68378 | 0.70133 | 0.61729 | 0.79633 | **0.80378** | 0.64363 | 0.80028 | 0.67254 | 0.67038 | 0.70198 | 0.70079 | **0.8089** |
| | | 0.8 | 0.69017 | 0.68699 | 0.70531 | 0.6187 | 0.79352 | 0.80449 | 0.64186 | 0.79983 | 0.66845 | 0.66638 | 0.70871 | 0.70385 | **0.81385** |
| | | 0.9 | 0.68133 | 0.68808 | 0.70735 | 0.61919 | 0.79654 | 0.80126 | 0.64454 | 0.80132 | 0.668 | 0.67294 | 0.70522 | 0.69572 | **0.8182** |
| | EU-Core | 0.7 | 0.78937 | 0.80496 | 0.79425 | 0.58671 | **0.92006** | 0.90148 | 0.60026 | 0.8577 | 0.83381 | 0.77581 | 0.81425 | 0.81109 | **0.92074** |
| | | 0.8 | 0.79647 | 0.80005 | 0.79958 | 0.58157 | **0.91658** | 0.89618 | 0.59322 | 0.85756 | 0.83258 | 0.7761 | 0.81998 | 0.81444 | **0.91708** |
| | | 0.9 | 0.79225 | 0.80729 | 0.80169 | 0.58084 | 0.91857 | 0.90959 | 0.59894 | 0.85782 | 0.82946 | 0.78051 | 0.81998 | 0.81493 | **0.9234** |
| | FB-Forum | 0.7 | 0.5172 | 0.51853 | 0.53271 | 0.5893 | 0.86763 | 0.86775 | 0.61829 | 0.84414 | 0.85318 | 0.73465 | 0.52119 | 0.54372 | **0.87494** |
| | | 0.8 | 0.5136 | 0.51497 | 0.52642 | 0.5845 | 0.86891 | 0.87138 | 0.62132 | 0.84515 | 0.85331 | 0.73297 | 0.52496 | 0.54278 | **0.88175** |
| | | 0.9 | 0.51622 | 0.51654 | 0.52327 | 0.59468 | 0.8685 | 0.87137 | 0.62045 | 0.84765 | 0.85446 | 0.72805 | 0.52582 | 0.54513 | **0.88497** |
| | CollegeMsg | 0.7 | 0.50149 | 0.50509 | 0.51621 | 0.56238 | 0.62795 | 0.63372 | 0.55942 | 0.61721 | 0.63924 | 0.59321 | 0.51307 | 0.52254 | **0.64524** |
| | | 0.8 | 0.5036 | 0.50509 | 0.51184 | 0.56643 | 0.62889 | 0.6382 | 0.55283 | 0.61886 | 0.63701 | 0.59599 | 0.51166 | 0.52112 | **0.64652** |
| | | 0.9 | 0.5034 | 0.5008 | 0.51003 | 0.56426 | 0.63191 | **0.64423** | 0.56529 | 0.61447 | **0.64063** | 0.58947 | 0.50801 | 0.5164 | 0.63159 |
| AUPR | MIT | 0.7 | 0.61856 | 0.62657 | 0.58372 | 0.65081 | **0.7533** | **0.75125** | 0.69668 | 0.72894 | 0.6811 | 0.65258 | 0.61367 | 0.64471 | **0.7558** |
| | | 0.8 | 0.62577 | 0.63108 | 0.58978 | 0.65737 | 0.75653 | 0.75933 | 0.70149 | 0.71742 | 0.68969 | 0.66921 | 0.62766 | 0.63832 | **0.77272** |
| | | 0.9 | 0.64995 | 0.62565 | 0.58856 | 0.6946 | 0.73856 | 0.7588 | 0.70733 | 0.7594 | 0.68812 | 0.67731 | 0.63718 | 0.63394 | **0.77975** |
| | Radoslaw-Email | 0.7 | 0.59542 | 0.59103 | 0.61263 | 0.49706 | **0.77314** | 0.75945 | 0.54831 | 0.75521 | 0.57998 | 0.57296 | 0.60723 | 0.60397 | **0.76671** |
| | | 0.8 | 0.60057 | 0.59979 | 0.60843 | 0.50377 | **0.77067** | 0.75823 | 0.54519 | 0.75779 | 0.56532 | 0.57302 | 0.61412 | 0.61214 | **0.77533** |
| | | 0.9 | 0.58284 | 0.58895 | 0.62165 | 0.50831 | **0.78153** | 0.76578 | 0.53343 | 0.76226 | 0.57251 | 0.57914 | 0.61465 | 0.60432 | 0.76623 |
| | EU-Core | 0.7 | 0.69562 | 0.72425 | 0.71433 | 0.4522 | **0.87381** | 0.83612 | 0.48384 | 0.76805 | 0.75092 | 0.68057 | 0.7407 | 0.73505 | 0.864 |
| | | 0.8 | 0.70928 | 0.7085 | 0.71033 | 0.45097 | **0.87006** | 0.82498 | 0.47696 | 0.76668 | 0.74148 | 0.6812 | 0.74712 | 0.73785 | **0.86033** |
| | | 0.9 | 0.70437 | 0.71942 | 0.71705 | 0.44668 | **0.8729** | 0.84576 | 0.48576 | 0.77288 | 0.74493 | 0.69552 | 0.74303 | 0.74207 | **0.86707** |
| | FB-Forum | 0.7 | 0.35425 | 0.33085 | 0.3862 | 0.46324 | 0.83518 | 0.83031 | 0.51946 | 0.79755 | 0.81527 | 0.64344 | 0.33775 | 0.4199 | **0.84107** |
| | | 0.8 | 0.33421 | 0.29424 | 0.37647 | 0.44884 | 0.83652 | 0.82829 | 0.51265 | 0.80158 | 0.81798 | 0.64254 | 0.38288 | 0.41241 | **0.84566** |
| | | 0.9 | 0.37623 | 0.31956 | 0.36049 | 0.48122 | 0.83917 | 0.83008 | 0.52914 | 0.80113 | 0.82186 | 0.64103 | 0.38164 | 0.4244 | **0.85126** |
| | CollegeMsg | 0.7 | 0.23637 | 0.22203 | 0.3241 | 0.40792 | **0.62043** | 0.57924 | 0.42707 | 0.55436 | 0.58719 | 0.48055 | 0.29945 | 0.37451 | 0.60555 |
| | | 0.8 | 0.26532 | 0.2379 | 0.28278 | 0.41916 | **0.63069** | 0.58834 | 0.42651 | 0.55535 | 0.59151 | 0.47761 | 0.30792 | 0.35818 | 0.59167 |
| | | 0.9 | 0.34346 | 0.20518 | 0.31656 | 0.42482 | **0.64382** | 0.60645 | 0.46551 | 0.54982 | 0.58214 | 0.49381 | 0.30351 | 0.3235 | 0.5755 |
| AVG PREC | MIT | 0.7 | 0.42537 | 0.43157 | 0.39182 | 0.45982 | **0.58591** | 0.57919 | 0.51131 | 0.54984 | 0.49088 | 0.45954 | 0.41925 | 0.45232 | **0.58722** |
| | | 0.8 | 0.43037 | 0.43916 | 0.39798 | 0.46354 | 0.58774 | 0.58906 | 0.51874 | 0.53464 | 0.50337 | 0.47835 | 0.43566 | 0.44652 | **0.60876** |
| | | 0.9 | 0.45532 | 0.43541 | 0.40031 | 0.51193 | 0.56143 | 0.59246 | 0.52812 | 0.59109 | 0.50277 | 0.49069 | 0.44881 | 0.43727 | **0.61778** |
| | Radoslaw-Email | 0.7 | 0.39214 | 0.38777 | 0.41128 | 0.30598 | **0.60355** | 0.58925 | 0.34576 | 0.58316 | 0.37677 | 0.36991 | 0.40612 | 0.40218 | **0.59965** |
| | | 0.8 | 0.39825 | 0.39751 | 0.40649 | 0.30913 | 0.59983 | 0.58806 | 0.34295 | 0.586 | 0.36302 | 0.37033 | 0.41356 | 0.41081 | **0.6126** |
| | | 0.9 | 0.38106 | 0.3865 | 0.42109 | 0.31426 | **0.6147** | 0.59841 | 0.3354 | 0.5931 | 0.37075 | 0.37675 | 0.41516 | 0.4016 | 0.59866 |
| | EU-Core | 0.7 | 0.5023 | 0.53955 | 0.52634 | 0.24898 | **0.76473** | 0.70274 | 0.27066 | 0.59821 | 0.57625 | 0.48325 | 0.56222 | 0.55393 | 0.74876 |
| | | 0.8 | 0.52078 | 0.51921 | 0.52146 | 0.24584 | **0.75977** | 0.68466 | 0.26353 | 0.59581 | 0.56178 | 0.4846 | 0.5707 | 0.55766 | 0.74298 |
| | | 0.9 | 0.51457 | 0.53344 | 0.53122 | 0.24556 | **0.7647** | 0.718 | 0.27395 | 0.60572 | 0.56923 | 0.50377 | 0.56512 | 0.56457 | 0.75402 |
| | FB-Forum | 0.7 | 0.18292 | 0.1834 | 0.19669 | 0.25585 | 0.70126 | 0.69385 | 0.2977 | 0.64301 | 0.66964 | 0.43773 | 0.186 | 0.21032 | **0.71197** |
| | | 0.8 | 0.18347 | 0.17313 | 0.1929 | 0.25111 | 0.70401 | 0.69105 | 0.29453 | 0.64921 | 0.6735 | 0.43607 | 0.19278 | 0.20784 | **0.71973** |
| | | 0.9 | 0.18288 | 0.18402 | 0.18402 | 0.26743 | 0.70702 | 0.69366 | 0.30285 | 0.64888 | 0.67904 | 0.43463 | 0.19464 | 0.21365 | **0.7286** |
| | CollegeMsg | 0.7 | 0.16825 | 0.17095 | 0.18361 | 0.22555 | 0.34653 | 0.33308 | 0.22299 | 0.31109 | 0.34401 | 0.26243 | 0.17577 | 0.19091 | **0.35845** |
| | | 0.8 | 0.16506 | 0.17393 | 0.17196 | 0.22227 | **0.35483** | 0.34407 | 0.21719 | 0.31268 | 0.34736 | 0.26465 | 0.18405 | 0.18352 | **0.35278** |
| | | 0.9 | 0.17149 | 0.16625 | 0.17282 | 0.2365 | **0.36697** | **0.36114** | 0.24277 | 0.30552 | 0.34299 | 0.27064 | 0.17635 | 0.19232 | 0.33737 |

TABLE 6.4: Performance comparison of $COMMLP - FULL$ with individual feature based link prediction algorithms using Linear Discriminant Analysis model on five datasets, three *Ratio* values and three performance metrics

| METRIC | DATASET | RATIO | CN | AA | JC | PA | SP | COSP | ACT | MFI | L3 | LOCALP | CCLP | NLC | COMMLP-FULL |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AUC | MIT | 0.7 | 0.82585 | 0.84117 | 0.76766 | 0.86451 | 0.84115 | 0.87684 | 0.81001 | 0.63006 | 0.87985 | 0.85749 | 0.82117 | 0.82339 | **0.9237** |
| | | 0.8 | 0.82605 | 0.83416 | 0.76121 | 0.8601 | 0.83954 | 0.88043 | 0.80465 | 0.64 | 0.88627 | 0.86325 | 0.83021 | 0.81765 | **0.92377** |
| | | 0.9 | 0.82555 | 0.84221 | 0.76833 | 0.86035 | 0.83997 | 0.87972 | 0.81202 | 0.64685 | 0.88085 | 0.85597 | 0.84013 | 0.82629 | **0.92442** |
| | Radoslaw-Email | 0.7 | 0.85725 | 0.86047 | 0.85423 | 0.82 | 0.87084 | 0.73566 | 0.7802 | 0.81013 | 0.86073 | 0.85526 | 0.85769 | 0.86086 | **0.91881** |
| | | 0.8 | 0.85915 | 0.86238 | 0.85321 | 0.82034 | 0.86256 | 0.73344 | 0.77525 | 0.82874 | 0.86301 | 0.85303 | 0.86032 | 0.86358 | **0.91173** |
| | | 0.9 | 0.86132 | 0.86259 | 0.85381 | 0.82626 | 0.86914 | 0.74157 | 0.77389 | 0.81835 | 0.86496 | 0.85341 | 0.85594 | 0.85492 | **0.92276** |
| | EU-Core | 0.7 | 0.9201 | 0.92509 | 0.92015 | 0.75677 | **0.9718** | 0.8287 | 0.76508 | 0.89425 | 0.94204 | 0.92342 | 0.93817 | 0.93941 | **0.97231** |
| | | 0.8 | 0.92239 | 0.93251 | 0.92384 | 0.76414 | **0.97241** | 0.81331 | 0.76504 | 0.86746 | 0.94153 | 0.92155 | 0.93401 | 0.93848 | **0.97294** |
| | | 0.9 | 0.92057 | 0.9251 | 0.91289 | 0.75414 | **0.97252** | 0.84774 | 0.76443 | 0.8665 | 0.94492 | 0.92278 | 0.93587 | 0.94055 | **0.97293** |
| | FB-Forum | 0.7 | 0.56669 | 0.57691 | 0.55592 | 0.73199 | 0.79283 | 0.82629 | 0.68392 | 0.80942 | 0.90553 | 0.82084 | 0.57554 | 0.60226 | **0.93931** |
| | | 0.8 | 0.56371 | 0.57259 | 0.55245 | 0.7353 | 0.80304 | 0.81997 | 0.68262 | 0.83943 | 0.90047 | 0.7967 | 0.57451 | 0.59884 | **0.93993** |
| | | 0.9 | 0.56311 | 0.5662 | 0.56231 | 0.72353 | 0.79874 | 0.83882 | 0.6896 | 0.82039 | 0.90803 | 0.81175 | 0.59639 | 0.61221 | **0.93019** |
| | CollegeMsg | 0.7 | 0.54278 | 0.53641 | 0.49003 | 0.62852 | 0.53916 | 0.64334 | 0.54031 | 0.64376 | 0.68165 | 0.64657 | 0.53342 | 0.52948 | **0.73274** |
| | | 0.8 | 0.54151 | 0.54124 | 0.49912 | 0.62096 | 0.54701 | 0.63904 | 0.5314 | 0.64282 | 0.67858 | 0.64582 | 0.5285 | 0.52808 | **0.75117** |
| | | 0.9 | 0.52677 | 0.54228 | 0.51721 | 0.62669 | 0.53137 | 0.62969 | 0.5321 | 0.63562 | 0.67447 | 0.63633 | 0.52525 | 0.52381 | **0.73045** |
| AUPR | MIT | 0.7 | 0.55525 | 0.54862 | 0.43035 | 0.61586 | 0.48342 | 0.66293 | 0.50735 | 0.31599 | 0.65477 | 0.61451 | 0.54242 | 0.54953 | **0.76082** |
| | | 0.8 | 0.54293 | 0.55851 | 0.4072 | 0.62223 | 0.45099 | 0.66204 | 0.50446 | 0.31278 | 0.67521 | 0.61086 | 0.56226 | 0.55037 | **0.76962** |
| | | 0.9 | 0.5702 | 0.56059 | 0.43063 | 0.61649 | 0.50002 | 0.67127 | 0.51333 | 0.34461 | 0.65161 | 0.60216 | 0.56464 | 0.5651 | **0.77336** |
| | Radoslaw-Email | 0.7 | 0.56784 | 0.56612 | 0.54757 | 0.51241 | 0.48088 | 0.47295 | 0.36645 | 0.47181 | 0.60176 | 0.58414 | 0.57729 | 0.6034 | **0.79372** |
| | | 0.8 | 0.57262 | 0.57218 | 0.55049 | 0.5191 | 0.47338 | 0.4741 | 0.36133 | 0.50033 | 0.6069 | 0.57583 | 0.58702 | 0.6013 | **0.78078** |
| | | 0.9 | 0.56717 | 0.59059 | 0.55651 | 0.53716 | 0.48147 | 0.47964 | 0.35851 | 0.46878 | 0.62455 | 0.58352 | 0.56491 | 0.59207 | **0.8087** |
| | EU-Core | 0.7 | 0.70991 | 0.7331 | 0.68824 | 0.43435 | 0.82276 | 0.69888 | 0.43023 | 0.68388 | 0.75807 | 0.71042 | 0.76307 | 0.76927 | **0.84187** |
| | | 0.8 | 0.71527 | 0.75024 | 0.70679 | 0.44066 | 0.81917 | 0.69705 | 0.43029 | 0.6831 | 0.75793 | 0.70699 | 0.76095 | 0.77223 | **0.84141** |
| | | 0.9 | 0.70796 | 0.72961 | 0.68223 | 0.4542 | 0.787 | 0.72549 | 0.43213 | 0.6683 | 0.76778 | 0.6884 | 0.76681 | 0.77966 | **0.85431** |
| | FB-Forum | 0.7 | 0.25174 | 0.251 | 0.19657 | 0.42175 | 0.3424 | 0.72101 | 0.25443 | 0.66915 | 0.80037 | 0.51953 | 0.25702 | 0.29221 | **0.86576** |
| | | 0.8 | 0.2499 | 0.24639 | 0.19682 | 0.42717 | 0.34833 | 0.7227 | 0.25214 | 0.6992 | 0.79553 | 0.48551 | 0.25659 | 0.29552 | **0.85971** |
| | | 0.9 | 0.25664 | 0.24117 | 0.21211 | 0.40687 | 0.35185 | 0.7455 | 0.26171 | 0.69138 | 0.81076 | 0.51467 | 0.28017 | 0.31823 | **0.85935** |
| | CollegeMsg | 0.7 | 0.22701 | 0.22914 | 0.18185 | 0.31582 | 0.23764 | 0.4425 | 0.28707 | 0.42875 | 0.48416 | 0.34433 | 0.2317 | 0.23789 | **0.52004** |
| | | 0.8 | 0.21868 | 0.22851 | 0.17817 | 0.32066 | 0.21682 | 0.44081 | 0.29044 | 0.41621 | 0.47883 | 0.3512 | 0.23239 | 0.24531 | **0.54481** |
| | | 0.9 | 0.21133 | 0.22447 | 0.18983 | 0.30952 | 0.23422 | 0.44581 | 0.30109 | 0.41885 | 0.48039 | 0.34854 | 0.23509 | 0.24159 | **0.49748** |
| AVG PREC | MIT | 0.7 | 0.56183 | 0.55444 | 0.43517 | 0.62146 | 0.55284 | 0.66589 | 0.51222 | 0.32132 | 0.65903 | 0.6196 | 0.5484 | 0.55537 | **0.76406** |
| | | 0.8 | 0.55096 | 0.56609 | 0.4144 | 0.62839 | 0.53121 | 0.66665 | 0.51098 | 0.32018 | 0.67976 | 0.61802 | 0.56987 | 0.55842 | **0.77489** |
| | | 0.9 | 0.58455 | 0.57481 | 0.44517 | 0.62872 | 0.5749 | 0.6815 | 0.5225 | 0.35633 | 0.66222 | 0.61566 | 0.5807 | 0.57819 | **0.78011** |
| | Radoslaw-Email | 0.7 | 0.56943 | 0.56779 | 0.54892 | 0.5141 | 0.63529 | 0.47361 | 0.36732 | 0.4734 | 0.60307 | 0.58582 | 0.57909 | 0.60501 | **0.79446** |
| | | 0.8 | 0.57509 | 0.57464 | 0.55254 | 0.52123 | 0.627 | 0.47528 | 0.36279 | 0.5026 | 0.6087 | 0.57831 | 0.58937 | 0.6037 | **0.78169** |
| | | 0.9 | 0.57174 | 0.59522 | 0.56077 | 0.54084 | 0.63615 | 0.48294 | 0.36143 | 0.47323 | 0.62745 | 0.58752 | 0.56956 | 0.59591 | **0.80973** |
| | EU-Core | 0.7 | 0.71092 | 0.73424 | 0.69006 | 0.43593 | 0.85957 | 0.7003 | 0.4326 | 0.68595 | 0.75905 | 0.71136 | 0.76406 | 0.77005 | **0.84331** |
| | | 0.8 | 0.71662 | 0.75135 | 0.70922 | 0.4433 | **0.85603** | 0.69931 | 0.43371 | 0.68571 | 0.7591 | 0.70879 | 0.76217 | 0.7734 | 0.84351 |
| | | 0.9 | 0.71095 | 0.73233 | 0.68728 | 0.45747 | 0.8424 | 0.72972 | 0.438 | 0.67288 | 0.7704 | 0.69167 | 0.76934 | 0.78178 | **0.85675** |
| | FB-Forum | 0.7 | 0.24239 | 0.24581 | 0.18929 | 0.42408 | 0.34875 | 0.71777 | 0.24438 | 0.67127 | 0.80107 | 0.52181 | 0.24738 | 0.28048 | **0.86551** |
| | | 0.8 | 0.24238 | 0.2419 | 0.19041 | 0.43058 | 0.35295 | 0.71988 | 0.24394 | 0.7023 | 0.79646 | 0.48876 | 0.24806 | 0.28433 | **0.86004** |
| | | 0.9 | 0.25063 | 0.24079 | 0.20683 | 0.41305 | 0.35927 | 0.74544 | 0.25737 | 0.69554 | 0.81206 | 0.51969 | 0.27283 | 0.30765 | **0.85976** |
| | CollegeMsg | 0.7 | 0.20903 | 0.21025 | 0.16717 | 0.31735 | 0.17477 | 0.42047 | 0.17782 | 0.42437 | 0.48091 | 0.34241 | 0.2174 | 0.22145 | **0.49995** |
| | | 0.8 | 0.2047 | 0.21321 | 0.17132 | 0.32395 | 0.17568 | 0.41866 | 0.17456 | 0.41443 | 0.47656 | 0.35 | 0.21772 | 0.2259 | **0.52198** |
| | | 0.9 | 0.20083 | 0.2116 | 0.17925 | 0.31778 | 0.18165 | 0.42469 | 0.18209 | 0.41911 | 0.48045 | 0.35028 | 0.22275 | 0.23093 | **0.48207** |

### 6.3.3 Performance comparison of $COMMLP-FULL$ with individual feature based methods using Linear Discriminant Analysis model

Table 6.4 compares the performance of the $COMMLP-FULL$ feature set with twelve individual link prediction algorithm-based feature sets on five datasets and three *Ratio* values on a Linear Discriminant Analysis-based classifier.

We have used default Scikit-Learn [156] implementation for this classifier.

With respect to the AUC performance metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets and *Ratio* values. The AUC values of individual feature-based link prediction algorithms are also quite high when compared with other machine learning models. In the AUPR performance metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets and *Ratio* values. Other algorithms which show decent performance in this metric are SP, L3, NLC, and CCLP for datasets MIT, Radoslaw-Email, and EU-Core. For FB-Forum and CollegeMsg, COSP and MFI show much better performance than SP, NLC and CCLP. In the Average Precision (AVG PREC) performance metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets except EU-Core. In EU-Core dataset, $COMMLP-FULL$ is slightly outperformed by SP for $Ratio = 0.7, 0.8$.

### 6.3.4 Performance comparison of $COMMLP-FULL$ with individual feature based methods using Random Forest Classifier model

Table 6.5 compares the performance of the $COMMLP-FULL$ feature set with twelve individual link prediction algorithm-based feature sets on five datasets and three *Ratio*

TABLE 6.5: Performance comparison of $COMMLP-FULL$ with individual feature based link prediction algorithms using Random Forest Classifier model on five datasets, three *Ratio* values and three performance metrics

| METRIC | DATASET | RATIO | CN | AA | JC | PA | SP | COSP | ACT | MFI | L3 | LOCALP | CCLP | NLC | COMMLP-FULL |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AUC | MIT | 0.7 | 0.82826 | 0.84372 | 0.82053 | 0.8724 | 0.89441 | 0.89028 | 0.89396 | 0.90318 | 0.88295 | 0.85952 | 0.8364 | 0.84868 | **0.92533** |
| | | 0.8 | 0.82681 | 0.84064 | 0.80723 | 0.88294 | 0.88805 | 0.89718 | 0.89157 | 0.90552 | 0.87809 | 0.86279 | 0.84691 | 0.85722 | **0.92986** |
| | | 0.9 | 0.81606 | 0.83909 | 0.82944 | 0.87925 | 0.90118 | 0.89015 | 0.90047 | 0.91399 | 0.88637 | 0.85852 | 0.83757 | 0.86743 | **0.94** |
| | Radoslaw-Email | 0.7 | 0.82074 | 0.84841 | 0.85408 | 0.79867 | 0.91419 | 0.91558 | 0.832 | 0.91672 | 0.85341 | 0.83867 | 0.84986 | 0.84696 | **0.93772** |
| | | 0.8 | 0.82148 | 0.84482 | 0.85324 | 0.80525 | 0.91474 | 0.91773 | 0.82865 | 0.91484 | 0.85397 | 0.83885 | 0.85106 | 0.85306 | **0.94186** |
| | | 0.9 | 0.82212 | 0.84647 | 0.85823 | 0.80223 | 0.91622 | 0.91624 | 0.83859 | 0.91758 | 0.85813 | 0.84387 | 0.85533 | 0.84774 | **0.94182** |
| | EU-Core | 0.7 | 0.8935 | 0.9163 | 0.9078 | 0.73266 | 0.97536 | 0.97159 | 0.75526 | 0.95831 | 0.94714 | 0.90919 | 0.92499 | 0.92495 | **0.98097** |
| | | 0.8 | 0.89308 | 0.91369 | 0.91324 | 0.73067 | 0.9752 | 0.97187 | 0.75384 | 0.9586 | 0.94396 | 0.91418 | 0.92395 | 0.92301 | **0.98339** |
| | | 0.9 | 0.89583 | 0.92046 | 0.90989 | 0.7318 | 0.97175 | 0.96914 | 0.75804 | 0.95909 | 0.94983 | 0.9136 | 0.92538 | 0.92295 | **0.98223** |
| | FB-Forum | 0.7 | 0.55701 | 0.5501 | 0.56008 | 0.7076 | **0.93117** | 0.9103 | 0.78233 | 0.91394 | 0.91731 | 0.85548 | 0.56308 | 0.56247 | **0.93572** |
| | | 0.8 | 0.57368 | 0.54523 | 0.555 | 0.70299 | 0.93004 | 0.91222 | 0.78399 | 0.91957 | 0.91404 | 0.86353 | 0.55063 | 0.56161 | **0.9405** |
| | | 0.9 | 0.56777 | 0.54918 | 0.55411 | 0.69042 | 0.93098 | 0.91274 | 0.80077 | 0.92176 | 0.91668 | 0.85542 | 0.56235 | 0.56129 | **0.94073** |
| | CollegeMsg | 0.7 | 0.53515 | 0.53471 | 0.53484 | 0.6389 | 0.72611 | 0.66045 | 0.65065 | 0.70258 | 0.70607 | 0.66367 | 0.52275 | 0.54383 | **0.75552** |
| | | 0.8 | 0.54722 | 0.52561 | 0.52869 | 0.63943 | 0.7275 | 0.65975 | 0.64344 | 0.69682 | 0.69775 | 0.66531 | 0.52441 | 0.52952 | **0.75862** |
| | | 0.9 | 0.55722 | 0.542 | 0.5342 | 0.649 | 0.72176 | 0.66707 | 0.64952 | 0.68658 | 0.69845 | 0.66617 | 0.52677 | 0.54511 | **0.76579** |
| AUPR | MIT | 0.7 | 0.58083 | 0.59824 | 0.56272 | 0.65367 | **0.75495** | 0.71128 | 0.69303 | 0.73485 | 0.64676 | 0.6372 | 0.56908 | 0.58674 | **0.75616** |
| | | 0.8 | 0.55334 | 0.59856 | 0.56215 | 0.66607 | 0.74486 | 0.73749 | 0.69025 | 0.72823 | 0.64375 | 0.65141 | 0.59281 | 0.6079 | **0.7641** |
| | | 0.9 | 0.56436 | 0.58318 | 0.55711 | 0.64769 | 0.76212 | 0.71312 | 0.70105 | 0.73541 | 0.6452 | 0.62492 | 0.54968 | 0.61886 | **0.79021** |
| | Radoslaw-Email | 0.7 | 0.53005 | 0.58994 | 0.59894 | 0.47878 | 0.8013 | 0.79301 | 0.54825 | 0.79103 | 0.58375 | 0.55348 | 0.60032 | 0.59532 | **0.83308** |
| | | 0.8 | 0.54329 | 0.57591 | 0.60128 | 0.49277 | 0.80005 | 0.79687 | 0.54118 | 0.78452 | 0.5912 | 0.55845 | 0.60243 | 0.59977 | **0.83541** |
| | | 0.9 | 0.53913 | 0.58605 | 0.60977 | 0.48858 | 0.81123 | 0.8 | 0.58516 | 0.79094 | 0.59279 | 0.57056 | 0.613 | 0.59449 | **0.82705** |
| | EU-Core | 0.7 | 0.66137 | 0.71513 | 0.69045 | 0.39594 | 0.86078 | 0.87018 | 0.44809 | 0.79843 | 0.75925 | 0.67436 | 0.74617 | 0.753 | **0.89365** |
| | | 0.8 | 0.66048 | 0.7119 | 0.70604 | 0.40381 | 0.87297 | 0.86613 | 0.43846 | 0.79308 | 0.7548 | 0.66272 | 0.7409 | 0.74365 | **0.91167** |
| | | 0.9 | 0.67416 | 0.72299 | 0.68524 | 0.40568 | 0.87914 | 0.86565 | 0.42694 | 0.81177 | 0.7718 | 0.68361 | 0.74634 | 0.76051 | **0.89967** |
| | FB-Forum | 0.7 | 0.23172 | 0.22595 | 0.25411 | 0.38886 | 0.84902 | 0.82291 | 0.48694 | 0.81431 | 0.81314 | 0.63352 | 0.24172 | 0.26477 | **0.86421** |
| | | 0.8 | 0.24528 | 0.22946 | 0.25685 | 0.38376 | 0.84476 | 0.82786 | 0.4962 | 0.81663 | 0.81703 | 0.65847 | 0.24137 | 0.26441 | **0.87447** |
| | | 0.9 | 0.24346 | 0.22511 | 0.25295 | 0.36995 | 0.83884 | 0.82692 | 0.51375 | 0.81587 | 0.80826 | 0.63877 | 0.24504 | 0.27074 | **0.86786** |
| | CollegeMsg | 0.7 | 0.20493 | 0.20481 | 0.2114 | 0.30292 | 0.48531 | 0.44749 | 0.30954 | 0.46346 | 0.48017 | 0.37104 | 0.21494 | 0.23799 | **0.53554** |
| | | 0.8 | 0.22372 | 0.19643 | 0.21476 | 0.31118 | 0.49435 | 0.43884 | 0.31048 | 0.46208 | 0.46511 | 0.37466 | 0.21475 | 0.21307 | **0.54311** |
| | | 0.9 | 0.22942 | 0.21472 | 0.2102 | 0.31877 | 0.50243 | 0.46074 | 0.30273 | 0.43986 | 0.48535 | 0.36228 | 0.20357 | 0.25728 | **0.55284** |
| AVG PREC | MIT | 0.7 | 0.58173 | 0.59963 | 0.56367 | 0.65386 | 0.69898 | 0.70877 | 0.6923 | 0.73394 | 0.64892 | 0.63736 | 0.57052 | 0.59003 | **0.7503** |
| | | 0.8 | 0.55811 | 0.60184 | 0.56365 | 0.66771 | 0.70431 | 0.73645 | 0.69141 | 0.73188 | 0.64738 | 0.65213 | 0.59587 | 0.61035 | **0.76365** |
| | | 0.9 | 0.57269 | 0.58996 | 0.56504 | 0.65116 | 0.70003 | 0.71463 | 0.70764 | 0.74069 | 0.656 | 0.62977 | 0.55844 | 0.62787 | **0.79221** |
| | Radoslaw-Email | 0.7 | 0.52884 | 0.5877 | 0.59585 | 0.47703 | 0.77864 | 0.78863 | 0.54564 | 0.78604 | 0.58145 | 0.55178 | 0.59781 | 0.59252 | **0.83037** |
| | | 0.8 | 0.54208 | 0.57375 | 0.59884 | 0.49157 | 0.77748 | 0.79294 | 0.53907 | 0.77985 | 0.58919 | 0.55679 | 0.60004 | 0.59741 | **0.83314** |
| | | 0.9 | 0.54069 | 0.58459 | 0.60938 | 0.48948 | 0.77645 | 0.79635 | 0.58376 | 0.78675 | 0.59334 | 0.57153 | 0.61153 | 0.59365 | **0.82494** |
| | EU-Core | 0.7 | 0.65975 | 0.71164 | 0.68838 | 0.39384 | 0.87314 | 0.86666 | 0.44553 | 0.79411 | 0.75609 | 0.67068 | 0.74216 | 0.74947 | **0.89241** |
| | | 0.8 | 0.65952 | 0.71008 | 0.70362 | 0.40258 | 0.878 | 0.86195 | 0.43631 | 0.78976 | 0.75191 | 0.66095 | 0.73781 | 0.73979 | **0.91014** |
| | | 0.9 | 0.67427 | 0.72171 | 0.68461 | 0.40618 | 0.87227 | 0.86194 | 0.42739 | 0.80834 | 0.77039 | 0.68244 | 0.74356 | 0.75747 | **0.89876** |
| | FB-Forum | 0.7 | 0.22194 | 0.2153 | 0.24195 | 0.38805 | 0.84046 | 0.80762 | 0.48427 | 0.80795 | 0.80479 | 0.62944 | 0.22222 | 0.23492 | **0.86065** |
| | | 0.8 | 0.23433 | 0.21943 | 0.24475 | 0.38401 | 0.83778 | 0.81455 | 0.49506 | 0.81065 | 0.80926 | 0.65483 | 0.22541 | 0.23613 | **0.87126** |
| | | 0.9 | 0.23715 | 0.21814 | 0.24671 | 0.3718 | 0.83318 | 0.81393 | 0.5133 | 0.8107 | 0.80058 | 0.63731 | 0.23003 | 0.24555 | **0.86498** |
| | CollegeMsg | 0.7 | 0.18988 | 0.19091 | 0.19552 | 0.29639 | 0.46226 | 0.4052 | 0.30074 | 0.44311 | 0.45385 | 0.35544 | 0.19451 | 0.1989 | **0.51586** |
| | | 0.8 | 0.20278 | 0.18425 | 0.19876 | 0.30546 | 0.47235 | 0.39851 | 0.3045 | 0.44316 | 0.44113 | 0.36034 | 0.19634 | 0.18951 | **0.52022** |
| | | 0.9 | 0.20847 | 0.19906 | 0.19976 | 0.31644 | 0.48131 | 0.42192 | 0.29945 | 0.42518 | 0.46259 | 0.35211 | 0.18848 | 0.22819 | **0.53221** |

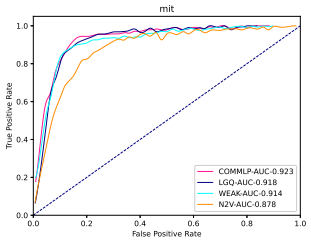
values on a Random Forest-based classifier. We have used 100 estimators as a setting to create this classifier and used the default Scikit-Learn implementation.

With respect to the AUC performance metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets and *Ratio* values. Some global similarity-based feature-based link prediction methods, i.e., SP, COSP, and MFI, show significantly better performance than other individual link prediction features. L3, which is a quasi-local similarity-based, is also a feature with exceptional performance, just behind the global similarity-based features. With respect to the AUPR metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets and *Ratio* values. Some global similarity-based feature-based link prediction methods, i.e., SP, COSP, and MFI, show significantly better performance than other individual link prediction features. L3, which is a quasi-local similarity-based, is also a feature with exceptional performance, just behind the global similarity-based features. In the Average Precision (AVG PREC) metric, our proposed $COMMLP-FULL$ feature set outperforms all other individual link prediction features for all datasets and *Ratio* values.

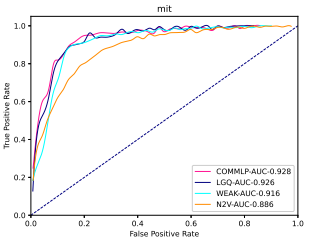
### 6.3.5 Comparison of Individual and Community Information based Link Prediction Features

In this section, we are going to discuss the three feature selection methods, $TREECL$, $KBMIR$ and $KBREG$ which we have used in this paper. Table 6.6 shows the features scores obtained by different link prediction features and community features. Initially, we experimented with twenty features. Out of twenty, twelve we have taken link prediction feature, like AA, JC, PA, CN, COSP, ACT, MFI, SP, L3, LOCALP, CCLP, NLC and eight community detection features such as DER, SURP, SBM, LEIDEN, SIGNI, CPM, EIGEN, and GREED. A large number of features is associated with computational complexity. More relevant features also lead to an increase in the overall

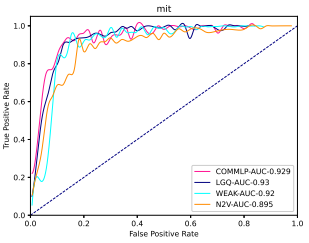
FIGURE 6.5: ROC curve based comparison of *COMMLP* (*COMMLP − DYN*) with other state-of-the-art feature based methods on Random Forest Classifier-based machine learning prediction



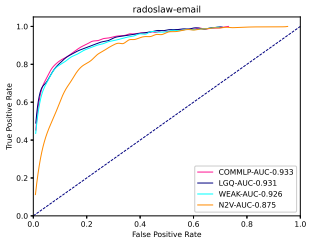

(A) MIT $R$ =0.5



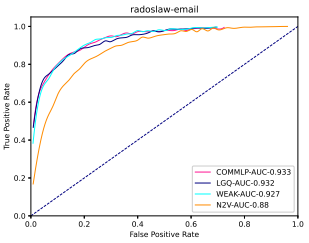

(B) MIT $R$ =0.7



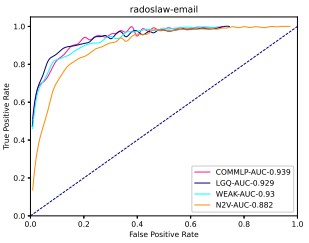

(C) MIT $R$ =0.9



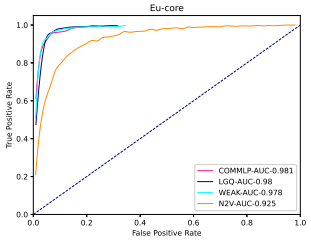

(D) Radoslaw-Email $R$ =0.5



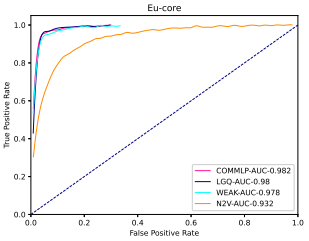

(E) Radoslaw-Email $R$ =0.7



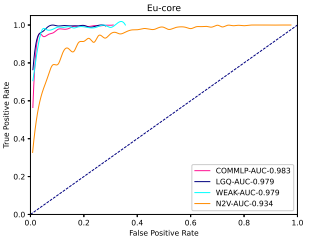

(F) Radoslaw-Email $R$ =0.9



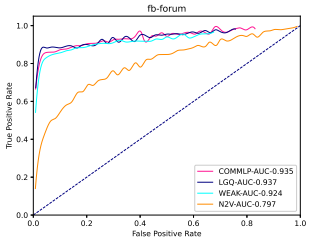

(G) EU-Core $R$ =0.5



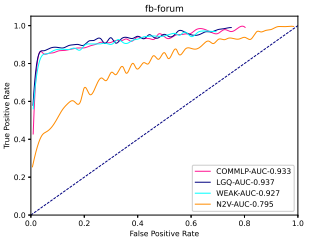

(H) EU-Core $R$ =0.7



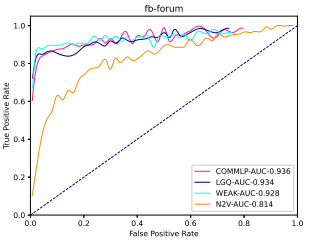

(I) EU-Core $R$ =0.9



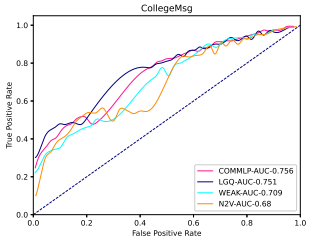

(J) FB-Forum $R$ =0.5



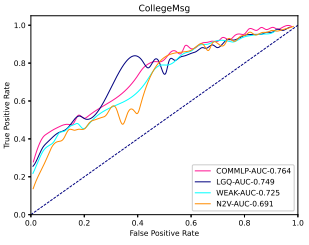

(K) FB-Forum $R$ =0.7



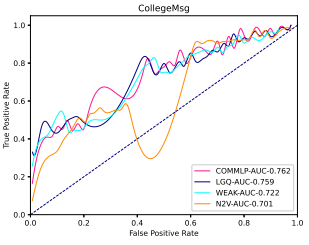

(L) FB-Forum $R$ =0.9




(M) CollegeMsg $R$ =0.5




(N) CollegeMsg $R$ =0.7




(O) CollegeMsg $R$ =0.9

TABLE 6.6: Features Scores of all individual features using TREECL, KBMIR and KBREG methods across all datasets

| METHOD | DATASET | CATEGORY | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Individual Link Pred | AA | JC | PA | CN | COSP | ACT | MFI | SP | L3 | LOCALP | CCLP | NLC |
| **TREECL** | MIT | | 0.4935 | 0.4229 | **0.7103** | **0.5111** | **0.6579** | 0.462 | 0.4634 | **1** | **0.9623** | **0.6993** | 0.468 | **0.5159** |
| | Radoslaw-Email | | **0.2468** | **0.2689** | 0.1939 | 0.2547 | **0.5376** | 0.1775 | **0.3354** | **1** | **0.3494** | **0.2906** | 0.2068 | 0.193 |
| | EU-Core | | **0.279** | **0.252** | 0.0829 | **0.2471** | **0.5054** | 0.083 | **0.2312** | **1** | **0.3837** | 0.225 | 0.2191 | 0.1679 |
| | FB-Forum | | 0.0572 | 0.0718 | **0.1457** | 0.0615 | **0.763** | **0.1391** | **0.4281** | **0.61** | **1** | **0.19** | 0.0323 | 0.0329 |
| | CollegeMsg | | 0.1265 | 0.1518 | **0.5201** | 0.1271 | **0.8218** | **0.4583** | **0.6619** | **0.5137** | **1** | **0.4194** | 0.1117 | 0.1178 |
| | | COMMLP Based | GREED | EIGEN | DER | SURP | SBM | LEIDEN | SIGNI | CPM | - | - | - | - |
| | MIT | | 0.2876 | **0.617** | **0.4517** | 0.3481 | **0.5816** | 0.2803 | 0.3342 | **0.3644** | | | | |
| | Radoslaw-Email | | 0.1933 | **0.372** | **0.2155** | 0.2093 | **0.2634** | **0.2134** | 0.1755 | 0.1997 | - | - | - | - |
| | EU-Core | | 0.111 | **0.1978** | **0.2006** | **0.3395** | **0.1805** | 0.1007 | 0.1254 | 0.1439 | - | - | - | - |
| | FB-Forum | | 0.1367 | **0.1559** | 0.1344 | **0.2084** | **0.1502** | 0.1452 | 0.1383 | **0.1683** | - | - | - | - |
| | CollegeMsg | | **0.4031** | 0.3561 | **0.3994** | **0.4614** | **0.3723** | 0.352 | 0.3407 | **0.3708** | - | - | - | - |
| | | Individual Link Pred | AA | JC | PA | CN | COSP | ACT | MFI | SP | L3 | LOCALP | CCLP | NLC |
| **KBMIR** | MIT | | **0.8512** | 0.7407 | **0.994** | 0.8007 | 0.7493 | **1** | **0.9052** | 0.7616 | **0.9843** | **0.8816** | **0.8327** | **0.8915** |
| | Radoslaw-Email | | **0.6894** | **0.6612** | 0.5667 | 0.6628 | **0.9821** | 0.6297 | **1** | **0.8424** | **0.6716** | 0.64 | **0.6871** | **0.6756** |
| | EU-Core | | **0.7591** | **0.7243** | 0.2729 | **0.7015** | **0.9954** | 0.29 | **0.9098** | **1** | **0.8183** | **0.7049** | **0.7799** | **0.7906** |
| | FB-Forum | | 0.0565 | 0.066 | **0.282** | 0.0434 | **0.9321** | **0.4051** | **0.8444** | **1** | **0.9001** | **0.5717** | 0.0538 | 0.0883 |
| | CollegeMsg | | 0.2153 | 0.25 | **0.5249** | 0.1937 | **0.9028** | **0.589** | **0.7965** | **1** | **0.9585** | **0.6613** | 0.213 | 0.2569 |
| | | COMMLP Based | GREED | EIGEN | DER | SURP | SBM | LEIDEN | SIGNI | CPM | - | - | - | - |
| | MIT | | **0.7672** | **0.8874** | **0.7904** | 0.7217 | **0.8986** | 0.5144 | 0.6874 | 0.7296 | - | - | - | - |
| | Radoslaw-Email | | 0.56 | **0.5841** | **0.6109** | **0.6153** | **0.6211** | 0.4907 | **0.5952** | **0.5853** | - | - | - | - |
| | EU-Core | | 0.4409 | **0.5559** | **0.5393** | **0.6684** | **0.5169** | 0.3042 | 0.4513 | 0.4899 | - | - | - | - |
| | FB-Forum | | **0.3054** | **0.3178** | **0.3552** | **0.3926** | 0.2949 | 0.2271 | 0.2996 | **0.3496** | - | - | - | - |
| | CollegeMsg | | 0.3441 | **0.4028** | **0.5233** | **0.4967** | **0.3877** | 0.2828 | 0.3738 | **0.3873** | - | - | - | - |
| | | Individual Link Pred | AA | JC | PA | CN | COSP | ACT | MFI | SP | L3 | LOCALP | CCLP | NLC |
| KBREG | MIT | | **0.6434** | 0.2551 | **0.8125** | **0.6783** | 0.0116 | 0.1743 | 0.0056 | **0.486** | **1** | **0.8426** | **0.512** | **0.5408** |
| | Radoslaw-Email | | **0.8579** | 0.5309 | **0.6149** | **0.9123** | 0.0479 | 0.2548 | 0.0051 | **0.5596** | **1** | **0.9753** | **0.562** | 0.3958 |
| | EU-Core | | **0.8917** | **0.7346** | 0.2229 | **0.8385** | 0.3887 | 0.048 | 0.0949 | **0.6341** | **1** | **0.7549** | **0.5919** | 0.405 |
| | FB-Forum | | 0.0131 | 0.0015 | **0.1724** | 0.0166 | **0.4314** | 0.0356 | **0.1611** | **0.1416** | **1** | **0.2908** | 0.0151 | 0.0099 |
| | CollegeMsg | | 0.0308 | 0.0028 | **0.1947** | 0.0339 | **0.522** | 0.0243 | **0.191** | 0.0404 | **1** | **0.3418** | 0.055 | 0.0457 |
| | | COMMLP Based | GREED | EIGEN | DER | SURP | SBM | LEIDEN | SIGNI | CPM | - | - | - | - |
| | MIT | | 0.2031 | **0.548** | 0.0064 | 0.0647 | **0.5829** | **0.1697** | 0.051 | 0.0597 | - | - | - | - |
| | Radoslaw-Email | | 0.2111 | **0.7489** | **0.3648** | **0.2469** | **0.7068** | **0.4259** | 0.1541 | 0.2434 | - | - | - | - |
| | EU-Core | | 0.316 | **0.5962** | **0.53** | **0.6526** | **0.5752** | 0.2813 | **0.3309** | 0.3399 | - | - | - | - |
| | FB-Forum | | 0.1422 | **0.2011** | 0.0716 | **0.2173** | **0.192** | **0.1665** | 0.1346 | 0.1664 | - | - | - | - |
| | CollegeMsg | | **0.246** | 0.2018 | **0.2222** | **0.3094** | **0.2696** | **0.2144** | 0.1155 | 0.1876 | - | - | - | - |

performance of predictions. Therefore, we have applied feature selection techniques to optimize the features based on their higher feature score.

In the MIT, Radoslaw-Email, Eu-Core, Fb-Forum, and CollegeMsg datasets, AA, CN, PA, MFI, COSP, SP, L3, and LOCALP have the highest feature scores for the Extra Trees Classifier (*TREECL*), Mutual information (*KBMIR*), and *f_regression* feature selection technique. Additionally, community features like SBM, EIGEN, SURP, and LEIDEN give the highest score across all datasets.

We have selected twelve optimized features - eight similarity-based link prediction methods e.g AA, CN, PA, MFI, COSP, SP, L3, LOCALP and four community information-based link prediction features, SBM, EIGEN, SURP and LEIDEN. The experimental result shows that the optimized features produce better results, and it is less complex and more efficient to create the reduced feature set. From this point on this

reduced and optimized feature set is referred to as $COMMLP-DYN$ ($COMMLP$ in tables and figures) and this will be the feature set using which we will evaluate our proposal against state-of-the-art algorithms.

### 6.3.6 Comparison of *COMMLP-DYN* with State-of-the-Art Methods after Optimization

Table 6.7 compares the performance of $COMMLP-DYN$ feature set, in combination with four different prediction machine learning models, with three other state-of-art algorithms. This comparison is done for three different performance metrics and on five datasets with respect to three *Ratio* values. The performance metrics are AUC, AUPR and Average Precision (AVG PREC). $COMMLP-DYN$ feature set is optimized form of $COMMLP-FULL$ feature set which is defined on the basis of feature optimization as mentioned in Section 6.3.5. The machine learning models used are Neural Network (NN), XGBoost (XGB), Logistic Regression (LR) and Random Forest Classifier (RFC). $COMMLP-DYN$ and $COMMLP$ have been used interchangeably from this point. In this table we also use an additional dataset Mathoverflow in addition to the five datasets already used for experimental analysis in this paper. This dataset is much larger than the other datasets (with respect to number of nodes and edges) and provides a better validation for application of our proposed method on larger graphs.

For the AUC metric, COMMLP-RFC has the best performance in all datasets and all three *Ratio* values except on the FB-Forum dataset, and COMMLP-LDA is the second-best performing algorithm on the same datasets. On the FB-Forum dataset, COMMLP-LDA is the best performing method, and COMMLP-RFC becomes the second-best. The performance of COMMLP-XGB and LGQ are comparable in most cases, and in MIT and Radoslaw-Email datasets, they are the third best-performing algorithms. WEAK is the third-best performing algorithm in EU-Core, FB-Forum, and Mathoverflow datasets, and N2V is the third-best performing algorithm in the

TABLE 6.7: Comparison of *COMMLP* (*COMMLP − DYN*) with state-of-the-art methods after optimization using truncated feature set on six datasets, 3 *Ratio* values and three performance metrics

| METRIC | DATASET | RATIO | LGQ | WEAK | N2V | COMMLP-NN | COMMLP-XGB | COMMLP-LDA | COMMLP-RFC |
|---|---|---|---|---|---|---|---|---|---|
| AUC | MIT | 0.7 | 0.82457 | 0.74640 | 0.68493 | 0.68090 | 0.82580 | 0.91682 | **0.92608** |
| | | 0.8 | 0.83272 | 0.80896 | 0.69057 | 0.66225 | 0.83594 | 0.91899 | **0.93222** |
| | | 0.9 | 0.83336 | 0.80606 | 0.68747 | 0.60387 | 0.84222 | 0.91307 | **0.94468** |
| | Radoslaw-Email | 0.7 | 0.80236 | 0.90524 | 0.76220 | 0.67049 | 0.81458 | 0.92109 | **0.93374** |
| | | 0.8 | 0.81328 | 0.91033 | 0.77295 | 0.71584 | 0.80979 | 0.92268 | **0.93196** |
| | | 0.9 | 0.80299 | 0.90818 | 0.77245 | 0.72748 | 0.80923 | 0.91898 | **0.93524** |
| | EU-Core | 0.7 | 0.92349 | 0.96852 | 0.94412 | 0.77109 | 0.91658 | 0.97380 | **0.98138** |
| | | 0.8 | 0.91646 | 0.97156 | 0.94527 | 0.76865 | 0.92003 | 0.97367 | **0.98071** |
| | | 0.9 | 0.91872 | 0.97142 | 0.94409 | 0.77202 | 0.91920 | 0.97315 | **0.98273** |
| | FB-Forum | 0.7 | 0.87610 | 0.91943 | 0.85460 | 0.69459 | 0.88105 | **0.94539** | 0.93826 |
| | | 0.8 | 0.88449 | 0.92775 | 0.85458 | 0.72479 | 0.88796 | **0.94912** | 0.93976 |
| | | 0.9 | 0.87626 | 0.92894 | 0.86869 | 0.69153 | 0.87132 | **0.94538** | 0.93025 |
| | CollegeMsg | 0.7 | 0.64675 | 0.66709 | 0.70704 | 0.57110 | 0.63702 | 0.73710 | **0.76734** |
| | | 0.8 | 0.63538 | 0.68110 | 0.70554 | 0.57086 | 0.64632 | 0.75737 | **0.77158** |
| | | 0.9 | 0.64934 | 0.69361 | 0.71619 | 0.59396 | 0.65169 | 0.75656 | **0.77455** |
| | Mathoverflow | 0.7 | 0.69930 | 0.73470 | 0.71503 | 0.69546 | 0.69930 | 0.72646 | **0.75179** |
| | | 0.8 | 0.69727 | 0.73450 | 0.71744 | 0.68851 | 0.69884 | 0.72620 | **0.75154** |
| | | 0.9 | 0.70007 | 0.73452 | 0.71513 | 0.69373 | 0.69686 | 0.72775 | **0.75190** |
| AUPR | MIT | 0.7 | 0.75740 | 0.53872 | 0.36867 | 0.60315 | **0.75377** | **0.75228** | 0.74281 |
| | | 0.8 | 0.75814 | 0.58337 | 0.36480 | 0.57784 | 0.75867 | 0.75899 | **0.79029** |
| | | 0.9 | 0.75234 | 0.56599 | 0.36802 | 0.52222 | 0.76503 | 0.74837 | **0.82569** |
| | Radoslaw-Email | 0.7 | 0.75921 | 0.70671 | 0.45262 | 0.45310 | 0.77061 | 0.79815 | **0.82331** |
| | | 0.8 | 0.76708 | 0.72562 | 0.47347 | 0.51195 | 0.77106 | 0.79043 | **0.81473** |
| | | 0.9 | 0.75536 | 0.72082 | 0.45347 | 0.50327 | 0.77305 | 0.79478 | **0.82125** |
| | EU-Core | 0.7 | 0.87210 | 0.81660 | 0.74446 | 0.61259 | 0.86206 | 0.85590 | **0.90539** |
| | | 0.8 | 0.85773 | 0.83603 | 0.75661 | 0.62116 | 0.86657 | 0.84522 | **0.89420** |
| | | 0.9 | 0.86247 | 0.84588 | 0.74345 | 0.61295 | 0.86475 | 0.84059 | **0.90308** |
| | FB-Forum | 0.7 | 0.84120 | 0.81673 | 0.65383 | 0.51525 | 0.85016 | 0.85595 | **0.86934** |
| | | 0.8 | 0.84979 | 0.83521 | 0.65677 | 0.55129 | 0.85237 | 0.86383 | **0.87306** |
| | | 0.9 | 0.83992 | 0.83014 | 0.67583 | 0.53498 | 0.83424 | 0.85789 | **0.86437** |
| | CollegeMsg | 0.7 | 0.58994 | 0.28444 | 0.40990 | 0.30339 | **0.59673** | 0.53671 | 0.52643 |
| | | 0.8 | 0.59497 | 0.26556 | 0.40954 | 0.31418 | **0.61853** | 0.51634 | 0.54185 |
| | | 0.9 | 0.57978 | 0.30956 | 0.42606 | 0.32818 | **0.58726** | 0.52342 | 0.53369 |
| | Mathoverflow | 0.7 | 0.66867 | 0.60224 | 0.48991 | 0.31525 | **0.67400** | 0.58255 | 0.63948 |
| | | 0.8 | 0.66826 | 0.60429 | 0.49383 | 0.32045 | **0.67157** | 0.58344 | 0.63899 |
| | | 0.9 | 0.66847 | 0.60445 | 0.48863 | 0.36679 | **0.67108** | 0.58532 | 0.63916 |
| AVG PREC | MIT | 0.7 | 0.58805 | 0.53375 | 0.37463 | 0.44116 | 0.58351 | 0.75259 | **0.75457** |
| | | 0.8 | 0.58729 | 0.58701 | 0.37293 | 0.38452 | 0.58754 | 0.76234 | **0.79146** |
| | | 0.9 | 0.58060 | 0.57146 | 0.38215 | 0.38785 | 0.59499 | 0.75500 | **0.82532** |
| | Radoslaw-Email | 0.7 | 0.58845 | 0.70808 | 0.45423 | 0.41578 | 0.60570 | 0.79852 | **0.82040** |
| | | 0.8 | 0.60070 | 0.72720 | 0.47600 | 0.47085 | 0.60598 | 0.79123 | **0.81199** |
| | | 0.9 | 0.58416 | 0.72333 | 0.45809 | 0.47891 | 0.60860 | 0.79582 | **0.81881** |
| | EU-Core | 0.7 | 0.76286 | 0.81778 | 0.74616 | 0.55060 | 0.74569 | 0.85720 | **0.90367** |
| | | 0.8 | 0.73858 | 0.83773 | 0.75853 | 0.58622 | 0.75388 | 0.84752 | **0.89251** |
| | | 0.9 | 0.74643 | 0.84840 | 0.74757 | 0.59356 | 0.75042 | 0.84383 | **0.90188** |
| | FB-Forum | 0.7 | 0.71131 | 0.81697 | 0.65568 | 0.44136 | 0.72628 | 0.85602 | **0.86523** |
| | | 0.8 | 0.72591 | 0.83552 | 0.65894 | 0.49220 | 0.73090 | **0.86344** | **0.86923** |
| | | 0.9 | 0.71072 | 0.83057 | 0.68043 | 0.49351 | 0.70175 | 0.85887 | **0.86060** |
| | CollegeMsg | 0.7 | 0.35874 | 0.34834 | 0.40491 | 0.23903 | 0.33975 | 0.48389 | **0.52941** |
| | | 0.8 | 0.34798 | 0.37843 | 0.42116 | 0.24405 | 0.35478 | 0.52044 | **0.53867** |
| | | 0.9 | 0.35052 | 0.42090 | 0.41345 | 0.27960 | 0.36840 | 0.52288 | **0.53153** |
| | Mathoverflow | 0.7 | 0.44033 | 0.53671 | 0.44281 | 0.31153 | 0.44433 | 0.52331 | **0.56364** |
| | | 0.8 | 0.43852 | 0.53695 | 0.44688 | 0.28714 | 0.44186 | 0.52400 | **0.56301** |
| | | 0.9 | 0.44086 | 0.53719 | 0.44154 | 0.30157 | 0.44031 | 0.52633 | **0.56373** |

CollegeMsg dataset. For the AUPR metric, COMMLP-RFC is the best performing algorithm in all datasets except CollegeMsg and Mathoverflow. Especially in MIT dataset for $Ratio = 0.7$, the performance of COMMLP-RFC is slightly worse than COMMLP-XGB, COMMLP-LDA and LGQ algorithms but it sees a significant performance increase for $Ratio = 0.8\&0.9$. COMMLP-XGB and COMMLP-LDA have comparable performance in MIT, EU-Core, and FB-Forum datasets. In the MIT dataset, LGQ also has comparable performance, and in the EU-Core dataset, it has the second-best performance out of all algorithms, just behind COMMLP-RFC. In CollegeMsg and Mathoverflow datasets, COMMLP-XGB has the best performance but is closely followed by the LGQ method. COMMLP-LDA has the third-best performance in the CollegeMsg dataset, and COMMLP-RFC has the third-best performance in the Mathoverflow dataset. For the Average Precision (AVG PREC) metric, COMMLP-RFC performs best across all datasets and all three *Ratio* values. COMMLP-LDA is the second-best performing algorithm in all datasets except the Mathoverflow dataset, where the WEAK algorithm slightly outperforms it. WEAK has decent performance in EU-Core and FB-Forum datasets also. COMMLP-XGB and LGQ have comparable performance in all datasets except the MIT dataset.

A point to note here is that different state-of-the-art algorithms have been proposed using various machine learning classification algorithms in their original research papers. To provide more uniform validation conditions, we fix Random Forest Classifier as the prediction algorithm for feature sets generated using COMMLP, LGQ, WEAK, and N2V algorithms. We perform link prediction using this approach on five datasets and present a graphical representation of the ROC-curve results in Fig.6.5. We run thirty iterations for each specific algorithm and each particular ratio. We also change the *Ratio* from $0.7, 0.8, \&0.9$ to $0.5, 0.7, \&0.9$ to gain a better understanding of algorithm performance changes with respect to incomplete information. The averaged-out AUC values for these experiments are listed in the graphs' legend, along with color-coded identifiers for each algorithm and their relevant ROC curves. The non uniformity of lines is because of averaging of ROC curves over multiple iterations of link prediction. We can observe

from graph legends that in terms of AUC value, our proposed *COMMLP* gives the best results in all cases of dataset and ratio combinations.

## 6.4 Conclusion

This work proposed a framework to generate community information-based link prediction features. These features enhance link prediction performance, typically conducted only with local, global, and quasi-local similarity-based features. Using feature relevance scoring, we demonstrated the superiority of our community features compared to some standard topological link prediction features. Finally, we provide an optimized feature set version of a combination of our community-based features with traditional link prediction-based features, $COMMLP - DYN$. This feature set performs better than other state-of-the-art algorithms for link prediction on dynamic networks in a snapshot-based setting. The experiments were conducted on six real-world datasets, three ratio training values to total edges, and three different performance metrics. In the future, this work can be extended using community detection methods that have been explicitly defined for dynamic graphs. Also, a new formulation can be researched on which takes into account overlapping community information rather than non-overlapping methods whose performance was discussed in this work.